PURDUE UNIVERSITY
SCHOOL OF AERONAUTICS AND ASTRONAUTICS

Interim Report:
A HIERARCHICAL APPROACH TO TARGET RECOGNITION AND TRACKING

Summary of Results for the Period April 1, 1989 - November 30, 1989

By Dominick Andrisani, II and M. Fernando Tenorio

February 7, 1990

Prepared for:
U.S. Army Research Office
Contract Number DAAL03-89-K-0086

West Lafayette, Indiana 47907

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) ARO 26813.1-EL
6a. NAME OF PERFORMING ORGANIZATION Purdue University 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION U. S. Army Research Office
6c. ADDRESS (City, State, and ZIP Code) West Lafayette, IN 47907	7b. ADDRESS (City, State, and ZIP Code) P. O. Box 12211 Research Triangle Park, NC 27709-2211
8a. NAME OF FUNDING / SPONSORING ORGANIZATION U. S. Army Research Office 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER DAAL03-89-K-0086
8c. ADDRESS (City, State, and ZIP Code) P. O. Box 12211 Research Triangle Park, NC 27709-2211	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
Interim Report: A Hierarchical Approach to Target Recognition and Tracking: Summary of Results for the period April 1, 1989 - November 30, 1989.

12. PERSONAL AUTHOR(S)
Dominick Andrisani, II and M. F. Tenorio

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Interim Technical	FROM ______ TO ______	1990/February 7	72

16. SUPPLEMENTARY NOTATION The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	

19. ABSTRACT

This research is aimed at studying a hierarchical target extraction, identification and tracking system based on passive sensors, that could be completely integrated with other battlefield resources. Involved with this is the study of a hierarchical structure for the mutually beneficial interconnection of multiple algorithms operating on several hierarchical levels. Together these simple algorithms would cooperate in the solution of a complex problem beyond the capability of any one algorithm.

Substantial progress at very modest cost (30,000 dollars) has been made in developing a passive hierarchical target identification and tracking system. A battlefield simulation capable of generating simulated images is under active development. With this simulation it is now possible to simulate images of a dynamic battlefield so that image processing and tracking algorithms can be studied. A new tracker for ground vehicles using position, attitude and terrain data has been specified. Artificial intelligence is being incorporated in two ways. First an intelligent predictor is being formulated. Second the high level reasoning module designed to use AI techniques to adjust and tune the various competing lower level modules is under development. (KR)

The image processing module was found to be able to operate below 0 dB with Gaussian additive noise, and can deliver throughput of over 8 times real-time image rates. A sophisticated software system that ties together each module of the hierarchical structure (CoHoRT) is being used and perfected that permit the easy modifiability of the system, testing of fault tolerant issues, and integration with previous AI efforts at ARDEC. New ways of looking at images from both multispectral and spatial viewpoints have been proposed, and preliminary results are promising.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code) / 22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

Interim Report:
A HIERARCHICAL APPROACH TO TARGET RECOGNITION AND TRACKING

Summary of results for the period: April 1, 1989 - November 30, 1989

By: Dominick Andrisani, II and M. Fernando Tenorio

February 7, 1990

Prepared for:
U.S. Army Research Office
Contract Number DAAL03-89-K-0086

Prepared at:
Purdue University
West Lafayette, In 47907

Approved for Public Release
Distribution Unlimited

The views, opinions, and/or findings contained in this report are those of the authors and should not be construed as official Department of the Army position, policy, or decision, unless so designated by other documentation.

TABLE OF CONTENTS

Interim Report:

A Hierarchical Approach to Target Recognition and Tracking

Summary of results for the period: April 1, 1989 - November 30, 1989

1. Abstract

This research is aimed at studying a hierarchical target extraction, identification and tracking system based on passive sensors, that could be completely integrated with other battlefield resources. Involved with this is the study of a hierarchical structure for the mutually beneficial interconnection of multiple algorithms operating on several hierarchical levels. Together these simple algorithms would cooperate in the solution of a complex problem beyond the capability of any one algorithm.

Substantial progress at very modest cost (30,000 dollars) has been made in developing a passive hierarchical target identification and tracking system. A battlefield simulation capable of generating simulated images is under active development. With this simulation it is now possible to simulate images of a dynamic battlefield so that image processing and tracking algorithms can be studied. A new tracker for ground vehicles using position, attitude and terrain data has been specified. Artificial intelligence is being incorporated in two ways. First an intelligent predictor is being formulated. Second the high level reasoning module designed to use AI techniques to adjust and tune the various competing lower level modules is under development.

The image processing module was found to be able to operate below 0 dB with Gaussian additive noise, and can deliver throughput of over 8 times realtime image rates. A sophisticated software system that ties together each module of the hierarchical structure (CoHoRT) is being used and perfected that permit the easy modifiability of the system, testing of fault tolerant issues, and itegration with previous AI efforts at ARDEC. New ways of looking at images from both multispectral and spatial viewpoints have been proposed, and preliminary results are promising.

2. Statement of the Problem Studied

This report describes a hierarchical target extraction, recognition, and tracking system based on passive sensors that consists of the mutually beneficial interconnection of multiple algorithms operating on several hierarchical levels. Together these algorithms or subsystems can cooperate in the solution of a complex problem beyond the capability of any one algorithm. This framework allows for easy addition, deletion or upgrading of new algorithms as well as communication between algorithms on the same and different levels. Furthermore this hierarchical framework provides extensive information to the users (at the highest level) to allow for performance assessment, human intervention, or operator training. Algorithms on different levels are allowed to deal with the world with different degrees of knowledge abstraction. Algorithms can compute a measure of confidence for each type of sensor output to aid in interpretation of results. Mechanisms for learning both within and between algorithms can be provided. It is felt that this hierarchical structure offers potential for robust operation since the loss of one or more algorithms would result in graceful degradation in the overall system performance.

This report also describes in greater detail a tracker algorithm that uses target orientation information from an image processor to track ground vehicles. The tracker will be utilized at the mid level of the hierarchical target recognition and tracking system. Furthermore, this report discusses image processing modules designed to detect, extract, and recognize targets from battlefield images.

The main philosophy behind hierarchical target extraction, identification and tracking system is that a typical single subsystem (sensor and related algorithms) has a performance versus effort relationship shown in Figure 1. To obtain maximum performance requires a very large effort. However, to obtain near maximum performance requires substantially less effort. Our approach is to design a cooperative system where multiple subsystems work together, and where each subsystem is operating at the "knee" of the performance versus effort curve. This approach recognizes that no one subsystem possesses sufficient performance, but with cooperation among the subsystems desirable overall performance is achievable. Furthermore, overall system robustness and reliability is enhanced while maximizing throughput. Each subsystem is simplified and the overall structure is parallelized for efficient implementation more suitable for real time operation.

In our ongoing research program we are developing the concepts described above, studying the implications of fusing the partial results of each algorithm with cooperating neighbors, and incorporating them in an hierarchical, completely passive, target identification and tracking system, capable of being integrated into a wider battle information network. The system is to be highly adaptive and capable of fusing information from a variety of sources (user-given, world, and sensed) in its decision making. The primary targets being considered are ground vehicles, although airborne targets can also be included. Pure imagery data will be used (possibly multiband), but the incorporation of laser range or radar data is also possible.

The system is composed of four hierarchical levels as shown in Figure 2.

1. Preprocessing Level: The sensing, filtering and distribution subsystems.

2. Low Level: Early processing; responsible for object extraction, tracking using simple image data (e.g. object centroid) and sensor pointing.

3. Mid Level: Secondary processing; responsible for object recognition, object orientation determination, and tracking using object orientation.

4. High Level: Reasoning; responsible for information fusion and coordination, user interaction, maintenance of world knowledge, and "smart" trajectory prediction.

At Purdue University a facility is being developed to implement and test this hierarchical structure. One of the key components in this effort is a battlefield simulation containing the moving targets and the moving tracker including the adjustable cameras and the images they generate. Those images can then be utilized as the primary raw sensor data for input to the hierarchical tracking system.

3. Summary of Most Important Results

This Interim Report summarizes results obtained during the period April 1,1989 to November 30,1989. The 30,000 dollars expended during this period is substantially less then that anticipated in our proposal[8].

3.1 A Justification for Image Based Trackers

On a modern battlefield enemy aircraft and ground vehicles will employ all the capabilities they can muster in order to guarantee their survival and their mission success. These capabilities involve speed, maneuverability, and unpredictability. There will be a substantial advantage to those vehicles that can move stealthfully without emitting radiation to reveal its position or intentions. There will be further advantage to any vehicle that can use the radiation emitted from an enemy vehicle to detect, recognize, track, and distroy that enemy.

These considerations suggest that the advantage will be held by any vehicle that uses passive (nonradiating) means to detect, recognize, and track and distroy the enemy.

To be effective the tracking system must be capable of tracking rapidly moving and maneuvering targets while themselves speeding along.

In this report we describe a class of hierarchical target tracking systems that are passive and capable of recognizing and tracking maneuvering targets. We employ video based imaging technology as the passive sensor and image processing to detect and track the target. Maneuvers of the target are detected and compensated for by extracting target aspect angle information from the images. This allows us to utilize the relationships that exist between ground vehicle aspect angles and vehicle linear velocity to more accurately determine target velocity. Knowing this we can more accurately predict future target motion, i.e. when firing at the target we can more accurately lead the target[2].

Target aspect angle information is a supplement to the sensor (camera or FLIR) pointing angle information that gives direct data as to the present position of the target. It is also proposed that sensor angular rates be employed to in order to obtain direct data on target velocity.

The Achilles heel of image based tracking lies in determining range to the target. We do not claim to have all the answers to this problem. We suggest the following approaches. A pulsed laser rangefinder can be used to determine range to target with great accuracy but at a loss of passive operation. Two separated sensors provide the ability to triangulate on the target or to use stereo imaging techniques to find range. Furthermore, it is also possible to borrow techniques employed by submarines to passively track other vehicles using only the bearing angle information provided by sonar sensors.

3.2 A Mid Level Tracker for Ground Vehicles

The key reason that image based target trackers offer the potential for superior performance over radar based trackers is that the image contains important information not included in radar or camera pointing data. That data is the target orientation (e.g. aspect angles or Euler angles).

In the case of fixed wing[3,4] or rotary wing aircraft[5,6] the orientation is directly related to target acceleration. For instance an aircraft turning will be banked into the turn. The amount of acceleration is proportional to the amount of banking. This new information helps the tracker to more accurately estimate target acceleration. This in turn helps the tracker make more accurate predictions of future target motion.

In contrast to aircraft, it is not possible to relate acceleration to orientation for tracked or wheeled vehicles. For example, a ground vehicle turning will generally stay level on the terrain surface even though the tires or tracks are generating side loads to curve the trajectory of the vehicle. Thus, in any single image a vehicle turning looks the same as a vehicle going straight (assuming that the imaging sensor is incapable of seeing the turning wheels).

On the other hand, it is possible to relate ground vehicle orientation to the direction of the velocity vector. This is because ground based vehicles roll over the surface causing the velocity vector to be in the direction that the vehicle is pointed. This is true with tracked vehicles and with wheeled vehicles. While some sliding of the wheels or tracks may frequently occur in battle, the primary mode of locomotion will be from rolling. Furthermore, the fact that rolling friction is greater then sliding friction and the necessity to maintain economy of fuel usage compels most drivers to roll rather then slide their tires or tracks.

In summary, the orientation of the ground vehicle gives direct information as to the direction of the velocity vector. It is believed that a tracker of ground vehicles that uses orientation information will outperform a tracker that does not.

As with aircraft, the change in vehicle orientation (heading angle) will precede a substantial change in cross track position as illustrated in Figure 3. This observation suggests increased accuracy in predicting (leading) the future position of the vehicle.

In this world of increasingly dense computer mass storage it is reasonable to assume that a next generation target tracker will have on-line access to a digital terrain map of the battlefield. This information can be used by the tracker in a number of ways.

1. Since the vehicle being tracked is constrained to move on the surface of the terrain, the terrain map can be used to help the tracker determine his present and future position. Certainly a vehicle approaching a steep decline can be predicted to loose elevation as he descends down the decline.

2. Since the vehicle will sit flat on the terrain, the orientation of the vehicle will be in part determined by the orientation of the terrain surface. This can be used to help determine the actual orientation of the target (i.e to aid the image processor) and to help track it.

3. The local slope of the terrain may be expected to influence the acceleration of the vehicle, e.g. a vehicle descending a hill will likely speed up.

4. Terrain data is also available with surface conditions, e.g. forests grasslands, marshes, roads. These also can be used by a "smart" tracker to predict future movements.

Tables 1 and 2 describe the state equations of a ground vehicle tracker that utilizes terrain maps in the first three ways mentioned above. Notice that if the tracker has some idea of the commanded speed, commanded heading turn rate, the tracker state equations, when integrated forward in time, will produce a trajectory that keeps the vehicle on the terrain surface at an orientation consistent with the terrain.

Table 3 describes one possible set of measurements that might be used by the tracker. Angles R_m, S_m, and T_m come from the image processor. This set assumes range, Ra, and range rate, $\dot{Ra}$, are available to the tracker from a laser rangefinder. This data may not be available if passive operation is required. In that case a second camera might be required to allow triangulation to find range. Angles η and ξ are the camera azimuth and elevation pointing angles respectively. Angular rates $\dot{\eta}$ and $\dot{\xi}$ are optional but useful pointing angular rates of the camera.

Equations of motion given in Table 2 are not designed to accurately simulate the motion of a ground vehicle. Instead, they are designed to track the motion of a ground vehicle given measurements of the motion of the vehicle. In the simulation process, it is important to accurately model all the forces and moments acting on the vehicle. However, in the tracking problem most of those forces and moments are unknown to the tracker. Thus, equations of motion for tracking can be much simpler then those for simulation.

Given the equations of motion and measurement equations as described above, we complete our tracker by utilizing an extended Kalman filter to estimate the present state of the target. In particular we use the "Continuous-Discrete Extended Kalman Filter" described in [7] and shown in Table 4. For predicting the future trajectory we integrate the nonlinear equations of motion in Table 2 forward in time assuming no process noise.

At the present time a tracker for ground vehicles of the type described above is being implemented at Purdue University for incorporation into the hierarchical tracking system at the mid level.

3.3 A Mid Level Tracker for Aircraft

We define a mid level aircraft tracker as a tracker that uses attitude data as well as position (radar) data to track the vehicle. While our emphasis in this research has been on tracking ground vehicles, the results we have recently obtained for aircraft are also applicable to ground vehicles.

We have developed several design methods that help reduce the sensitivity of our trackers to unknown knowledge about the vehicle being tracked. These results are described in Appendix C. The example given in Appendix C to illustrate the insensitive design methods is the $\alpha - \beta$ tracker, one of the choices possible for use in a low level tracker module.

Under subcontract with Applied Technology Associates, we have designed generic aircraft trackering software. As part of that study we have uncovered the explanation for conclusion 2 of page 8 of reference [9]. We quote that conclusion below

> "When an aircraft maneuver takes place primarily in the horizontal plane the generic filter (G.F.) outperforms the more complicated maneuver estimator (M.E.) as well as the center of gravity tracker (C.G.)"

This conclusion is based on using a prediction algorithm that allows the geometric angle of attack to change over the prediction interval (IFLAG(8)=0). A much superior predictor is the one that holds the geometric angle of attack constant over the prediction interval (IFLAG(8)=2). Using this superior predictor we have shown that the best results of reference [9] can be improved by 17%. These results are summarized below using data from page 6 of [9].

Total R.M.S. Prediction Error (ft.)

	M.E.	C.G.	G.F.	M.E.(with IFLAG(8)=2)
Right Turn	28.64	71.64	27.59	22.84

3.4 The Symbiosis Between Image Processing and Tracking

A fortunate benefit of optical target tracking systems is that by tightly linking the image processing subsystem and target tracking subsystem, a symbiotic relationship can exist between these two subsystems thereby improving overall tracker performance.

Specifically, the image processing subsystem enjoys the following benefits from its close association with the target tracking subsystem.

1. Accurate trac'.ng of the maneuvering target allows the optical sensor to employ a longer focal length thereby creating a larger target image within the field of view. In addition, accurate tracking helps eliminate loss of the target from the field of view.

2. Since a target usually is pointed into its velocity vector, the tracker (knowing the velocity vector) can help the image processor resolve the visual ambiguity associated with determining whether the target is pointing into or away from the imaging sensor.

3. The tracker can often provide a priori estimates of the location and aspect angle of the target within the sensor's field of view. This can reduce computer time required by many image processors, especially ones that use library search techniques to find target attitude.

The tracking subsystem enjoys the following benefits from its close association with the the image processing subsystem.

1. Location of the target within the image frame (i.e. boresight corrections) can be used to more accurately locate the target on the battlefield. This data supplements the sensor pointing data (i.e. the direction of the sensor boresight).

2. By knowing target aspect angle the tracker can more accurately estimate the direction of the target velocity vector and therefore determine with greater precision the present and future position of the target.

3.5 Low Level Image Processing Module

In this phase of the research, we have concentrated our efforts on the improvement of Markov Random Field based segmentation models for serial implementation and the development of new algorithms for parallel implementations. These types of algorithms can be divided into two separate phases: the initialization of the network, and the relaxation process that leads to the final interpretation. The network represents the pixels of the image (nodes) and the dependencies of these pixels according to geometric, pragmatic, syntactic, spatial and temporal constraints. In our first implementation, a simple segmentor was constructed including only spatial constraints. These are translated as links between nodes that are in an eight_nearest_neighbor (8-NN) neighborhood. The constraints symbolize the continuity of the objects that are sought against the background.

The initialization process defines the initial state of the network. It can take the form of a neighborhood or a single pixel sampling. Based on the homogeneity of the objects sought, the algorithm classifies the pixel either based on the neighborhood or its own value using a maximum likelihood and/or conditional probability function.

The relaxation process allows the network to use the previously initialized values, and through a descent in the space of segmented images, find the one that minimizes the continuity function given the likelihood values. This compromise process can be viewed as an optimization function performing a gradient descent or a stochastic search in the space. Gradient descents have been shown to be very effective for general classes of practical airplane imagery, but the stochastic search, although more expensive, led to good segmentation results even under signal-to-noise ratios of 0 dB, and less.

One experiment performed during this phase was the segmentation of an image using an 8-NN initialization function and an 8-NN continuity function with a stochastic search (Simulated Annealing). The image segmented had 4 levels of difference in the image value of the object versus the background. It was then submitted to about 200 level Gaussian white noise (50:1). The object was still acquired from the background. Although distorted, the object could easily be used to provide the center of gravity estimation for the tracker module.

Improvements in the algorithm have lead to the design of a faster serial algorithm performing 1/4 million pixel segmentation in under 20 seconds on a Sun 3/60. This corresponds to a rate of 12.5 pixels/millisecond segmented, or 75 pixels/millisecond analyzed. In a fully parallel hardware (single chip), the rate of segmented pixels can be in the order of 3,125 pixels/microsecond. This translates into a image rate of 2.08 images/millisecond (image of 512x512 pixels) or about 480 images segmented per second; 8 times faster than real-time tv pictures. One can detect objects that move a pixel in half a millisecond. This corresponds to speeds of 30 ft/sec at a distance of 4 miles.

Preliminary studies have demonstrated the feasibility of both digital and analog implementation of such a circuit. Analog circuits are capable of even faster implementations. Optronics (optico-electronic technology) could also provide a large speed up. The size of such a circuit can be made with current technology from a few down to a single chip, certainly within the acceptable limits in size, weight, and power. The hardest technical difficulty for the implementation of such circuits is the data input-output bottleneck to sustain these rates. Ideas such as the direct loading of the image onto the chip via optical means (such as done with CCD cameras), and data compression by representing segmented images by a few parameters seem promising. At Purdue, we have attempted the implementation of the entire circuit, and achieved a few pixel circuit implementation.

Although promising, this was not the main focus of the research, and limited funds led to the postponement of this direction.

The first level of processing should eventually include several resonating algorithms. At this point, this segmentor is being ported to our parallel distributed language for system integration. The low level processing is then restricted to a single algorithm.

3.6 Mid Level Image Processing: Identification of Object Parts

Experiments have been also made with the interpretation of binary segmented images using neural networks technology. By the use of supervised training algorithms, specially devised algorithms have been trained to recognize wire frame models distancing 5 degrees in 3-D view. Fixed structure networks can perform recognition with success rates of up to 91% in images of only 16 pixels across (a tank at over 4 mile distance). Variable structure networks have a clear advantage over fixed structure networks. A very restricted training at this point has produced over 98.3% recognition over the same type of data in distinguishing a T55 from a M113 with 72 different views. More tests should be done to confirm and improve these results using several wire frame model known techniques, such as homogeneous sampling.

The pursuit of this direction is a departure of the traditional type of design, and therefore outside the scope of this first phase of the research. Additional funding in this area is necessary to the thorough study of the implications of these algorithms to recognition.

3.7 Intermodule Communication: CoHoRT

During the first phase of this project, the Concurrent Hypermedia Reasoning Tool (CoHoRT) language has been ported to several machines on the ECN network. The reason for using CoHoRT is to produce a highly integrated passive target tracking and recognition system based on a fault tolerant self reconfiguring concept with several Sun 3/60 computers, the Ardent Titan and other supporting machines operating in a network.

Although CoHoRT on the Sun computers is mature, the port to the Ardent Titan machine will have to wait a more stable computing environment. The implementation of the distributed system is then being ported to a group of Suns, loosely tied to the Ardent for the time being.

CoHoRT will provide an environment for real-time fault test, parallel algorithm instrumentation, noise and fault introduction in the system, self reconfigurability, and quick system modification. Since our goal is passive target tracking and recognition, CoHoRT will provide a testbed for introduction of other types of sensors, as well as fusion of active information without extensive recoding. CoHoRT can also be used to incorporate ARDEC's AI based systems already developed into a single system, since it is capable of running rule based code.

3.8 The Battlefield Simulation Module

The approach taken to study the passive tracking problem involved a battlefield simulation that created computer generated imagery of a dynamically changing battlefield. Images of the battlefield generated by computer with an arbitrary degree of clarity can be sent to the image processing modules for target extraction, recognition and orientation determination. The results of this processing can then be sent to advanced target trackers which determine the present and future position of the target. Finally the tracker can send drive signals to the tracking camera to keep the target centered in the field of view. Furthermore the tracker can send drive signals to a fine control system for aiming of a cannon at the moving target off in the distance.

The simulation and image generation is being implemented on an Ardent Titan Graphics superminicomputer using C language and the Dore dynamic rendering language. This computer comes free of charge to this contract.

The present status of this work is as follows:

1. A simple mathematical model of a tank with steering and speed commands is implemented. An operator sitting at the console of the Ardent Titan can drive the tank using dials on the Titan knob panel.

2. A simple battlefield with several large and small objects on flat terrain is implemented.

3. The simulated camera viewing the battlefield can be commanded in azimuth, elevation, and focal length. The number of pixels of the image is easily adjustable. The camera can be adjusted either manually from the Titan knob panel or can be commanded by the tracker.

4. A dummy image processor module has been implemented to receive message data from the simulation and pass this unmodified to the tracker.

5. A simple centroid tracker has been implemented to keep the camera pointed at the tank.

6. Message formation for message passed from the simulation module to the image processor module has been specified.

7. Message format for the message passed from the image processor module to the tracker module has been specified.

8. Message format for the message passed from the tracker module back to the simulation module has been specified.

9. Simplified closed loop operation of the simulation, dummy image processor, and tracker to keep the target centered in the field of view has been demonstrated.

10. At present our algorithm requires that the image and image message be accompanied with range to target as might be known from a laser range finder. This may be changed if a second camera is available to allow triangulation on the target.

3.9 The Instrumentation Module

Determining exactly what is happening in a multi-module hierarchical system can be a difficult task without careful system design. In order to clearly display the status of all algorithms a separate module is being designed to display to the researcher/user the status of the various modules. This module will include various windows drawn using the X-window manager. Each window will display either graphically, with processed images or with text the status of one module. In this task we will use Athena Widgets Release 4.

It is hoped that this instrumentation module might in the future be expanded to allow for user interaction (manual intervention) with the hierarchical system. This capability would be a natural compliment to the reasoning capabilities designed into the high level reasoning module.

3.10 The Intelligent Predictor Module

The need for an intelligent predictor is based on the fact that the target vehicle operator is capable of suddenly performing large amplitude maneuvers that no conventional trajectory predictor (i.e. the optimal nonlinear predictor) is capable of following. However, often these maneuvers are not random, but are based on some mission objectives being pursued within the battlefield environment. If there were some way of determining his mission objectives and examining the battlefield environment it might be possible to predict more accurately the future behavior of the vehicle. An intelligent predictor that uses techniques from the literature of artificial intelligence might be able to predict future behavior better than a conventional predictor. A detailed discussion of this problem is given in Appendix E. Research is continuing at Purdue to develop these ideas for incorporation in the hierarchical target recognition and tracking system as a high level predictor to complement the conventional predictors associated with mid and low level tracking modules.

Let us consider the short term objective of a land vehicle in a battlefield environment. The driver has some mission objective that influences the general decisions made in controlling the vehicle to some destination. Physically, the vehicle is constrained to the surface of the earth. However, there are certain features of the terrain that the vehicle cannot or might not traverse. Rocks, ravines, lakes, trees, debris, high slope regions, and other obstacles are features maneuvered around during the course of driving the vehicle to some destination. The influence such environmental conditions have on the decisions made by the driver are a complex combination of human factors, driving experience, mission objectives, tactics, etc., and developing a relationship between the environmental causes and the drivers actions may not be feasible, if even possible. What is desired in intelligent prediction is to develop a model for a decision making program that, given the same environment, produces similar actions compared to the actual vehicles motion. The distinction here is that the decision making program may not necessarily make its decisions based on the same factors as the human driver, however, it is designed to produce decisions that are close to the actual drivers decisions.

The intelligent predictor uses "world data" to predict future motion. As describe above, the battlefield environment is complex; short term motion of the order of seconds may be influenced by visibility conditions, terrain features, mission objectives, tactics, intelligence reports, etc. In order to predict the future motion of the vehicle, one must model these types of data, and reason about how they influence the motion. The intelligent predictor can use a map model of the world (e.g., a DMA map database) to see what terrain the tracked vehicle will encounter in the future. It also has available to it intelligence information about possible strategic locations that the vehicle might be headed for as a destination. It is this type of "world data" that the intelligent predictor is trying to add to the estimation problem in order to intelligently bias the estimation of where the vehicle will be in the future.

3.11 The High Level Reasoning Module

Our research in the area of the high level supervisor has been greatly aided by the recent addition of Professor Edson Baptista to our resarch team. He is a visiting scholar from Brazil and has volunteered his efforts to this research for at least one year. His thoughts on the High Level Reasoning Module can be found in Appendix D.

The High Level Reasoning Module is responsible for information fusion and coordination, maintenance of world knowledge, and user interaction. We will attempt to add "human-like" capabilities to the system. In order to achieve this we will incorporate extensive data bases and knowledge processing into our system to simulate human expertise as well as provide mechanisms for direct human intervention.

More specifically, the High Level Reasoning Module (HLRM) is the one responsible for:

1. information fusion from then lower level algorithms and the intelligent predictor;
2. maintenance of world knowledge, enemy capabilities and tactics;
3. coordination of the lower level algorithms and judgement of their performances;
4. coordination of sensors performance as well as their focus of attention in order to extract extra identification information;
5. learning capability of target characteristics and tactics;
6. evaluation of threat and action sequence based on tactical rules and intelligence.

3.12 Experiments on Multiple Sensor Integration - Vision

The original segmentation schemes was also extended to the multidimensional multisensor case. In this mode, several wave lengths and types of aligned sensors can be fused to avoid noise and ghost images. This led us into the research of multispectra (Hi-D) data for battlefield target recognition. Most materials fall only in three distinct classes of reflectance characteristics. Metallic objects are very much unique in this fashion. By modeling the multispectra physical properties of these objects, and normalizing out brightness, one could create segmentation schemes that independent of camouflage, and outside lighting condition, and it is also immune to shadowing and other common image noise problems. The difficulty lies in determining the characteristics of reflectance in military structures.

Although our preliminary results look very promising, and the Purdue Laboratory for Application of Remote Sensing has the necessary equipment for this study, this laborious path will not be pursued any further for lack of personnel resources.

3.13 Ideas on Integration of Acoustic Sensing with Visual Sensing for Improved Tank Defense Against Helicopters

We have hypothesized that military ground vehicles, being vulnerable to attacks by helicopters, could use acoustic signatures and signal variation to determine position, attitude and type of helicopter in the battle field. With interest shown by General Dynamics on the problem (recent paper in Intelligent Systems Review), we feel even more confident that it is possible to adopt acoustic sensor arrays for ground vehicle defense. Our preliminary estimation shows that vehicle attitude can be hard to determine, but location and loose range estimation, with some degree of identification are possible with current technology.

4. List of All Publications

As a result of the work performed so far, we have produced the following papers:

a. D. Andrisani, M.F. Tenorio, J. Lu, and F.P. Kuhl, "A Hierarchical Approach to Passive Robust Target Tracking", presented at the Ninth Meeting of the Coordinating Group on Modern Control Theory, October 24-25, 1989, Picatinny Arsenal, NJ.

b. M.F. Tenorio, D. Andrisani, C. Codrington, and F.P. Kuhl, "Resonating Structures for Image Based Tracking", presented at the Ninth Meeting of the Coordinating Group on Modern Control Theory, October 24-25, 1989, Picatinny Arsenal, NJ.

c. D. Andrisani and E.T. Kim, "The Design of an Insensitive Estimator for Target Tracking", submitted for presentation at the 1990 AIAA Guidance, Navigation and Control Conference, August 20-22, 1990, Portland, Oregon.

5. Participating Personnel

Presented below are the key personnel working on this contract:

Professor Dominick Andrisani
Professor M. Fernando Tenorio
Professor Edson Baptista, Visiting Scholar from Universidade
Federal do Espirito Santo - Brazil
(no charge for his contributions)
Mr. E.T. Kim, Ph.D. student of Professor Andrisani
Mr. James Krozel, Ph.D. student of Professor Andrisani,
Hughes Fellowship recipient
(no charge for his contributions)
Mr. Jun Lu, Ph.D. student of Professor Andrisani
Mr. Craig Codrington, Ph.D. student of Professor Tenorio
Mr. Anthony Gibbens, M.S. student of Professor Tenorio
Mr. John A. Kassenbaum, Ph.D student of Professor Tenorio

6. Concluding Remarks

Substantial progress at very modest cost (30,000 dollars) has been made in developing a passive hierarchical target identification and tracking system. A battlefield simulation capable of generating simulated images is under active development. With this simulation it is now possible to simulate images of a dynamic battlefield so that image processing and tracking algorithms can be studied. A new tracker for ground vehicles using position, attitude and terrain data has been specified. Artificial intelligence is being incorporated in two ways. First an intelligent predictor is being formulated. Second the high level reasoning module designed to use AI techniques to adjust and tune the various competing lower level modules is under development.

The image processing module was found to be able to operate below 0 dB with Gaussian additive noise, and can deliver throughput of over 8 times realtime image rates. A sophisticated software system that ties together each module of the hierarchical structure (CoHoRT) is being used and perfected that permit the easy modifiability of the system, testing of fault tolerant issues, and integration with previous AI efforts at ARDEC. New ways of looking at images from both multispectral and spatial viewpoints have been proposed, and preliminary results are promising.

7. Bibliography

[1] Tenorio, M.F., Andrisani, D., Codrington, C., Kuhl, F.P., "Resonating Structures for Image Based Tracking", presented at the *Ninth Meeting of the Coordinating Group on Modern Control,* October 24-25, 1989, U.S. ARDEC, Picatinny Arsenal, Dover, NJ.

[2] Wrigley, W. and Hovorka, J. *Fire Control Principles,* McGraw-Hill Book Company, Inc. 1959.

[3] Andrisani, D., Kuhl, F.P., Gleason, D., "A Nonlinear Tracker Using Attitude Measurements," *IEEE Transactions on Aerospace and Electronic Systems,* Vol. AES-22, No. 5, September, 1985.

[4] Andrisani, D., Gorman, J.W., Mitchell, O.R., Thurman, S.W., "Orientation Identification and Maneuver Estimation of Tactical Targets Using Video Data," Final Report Battelle Delivery Order No. 0463, August 29, 1983. Prepared for Battelle Columbus Laboratories Scientific Services Program, Research Triangle Park, NC.

[5] Andrisani, D., Kuhl, F.P., Schierman, J.D., "Helicopter Tracking Using Attitude Measurements," *Proceedings of the Eighth Meeting of the Coordinating Group on Modern Control Theory,* Rochester, Michigan, February, 1987.

[6] Andrisani, D., Schierman, J.D., "Tracking Maneuvering Helicopters Using Attitude and Rotor Angle Measurements," *Conference Record: Twenty-First Annual Asilomar Conference on Signals, Systems and Computers,* Pacific Grove, California, November 2-4, 1987, pp. 328-333.

[7] Gelb, A., Editor, *Applied Optimal Estimation* M.I.T. Press, 1974.

[8] Tenorio, M.F., and Andrisani, D., "A Proposal Entitled 'A Hierarchical Approach to Target Recognition and Tracking'", submitted to the U.S. Army Research Office, December 1, 1989, revised February 23, 1989.

[9] Brown, R.E., and Goldberg, L., "A Comparative Study of a Generic Tracker", a report prepared for U. S. Army Armament Research, Development, and Engineering Center under Contract #DAAL03-86-D-0001, August 23, 1989.

8. Tables

Table 1
State Vector for Ground Vehicle Tracker

$\bar{x} =$	V_c	commanded linear velocity along the terrain surface
	V	actual linear velocity along the terrain surface
	x	North inertial position
	y	East inertial position
	z	down inertial position
	$\dot{R}_c$	commanded turn rate
	$\dot{R}$	actual turn rate
	R	heading angle
	S	pitch angle
	T	bank angle

Table 2
State Equations for Ground Vehicle Tracker

$$\dot{V}_C = w_1$$
$$\dot{V} = -(V - V_C)/\tau_V + g_v + w_2$$
$$\dot{x} = Vf_1(R,S,T)$$
$$\dot{y} = Vf_2(R,S,T)$$
$$\dot{z} = Vf_3(R,S,T) - [z - Z(x,y)]/\tau_z + w_5$$
$$\ddot{R}_c = w_6$$
$$\ddot{R} = -(\dot{R} - \dot{R}_c)/\tau_R + w_7$$
$$\dot{R} = \dot{R}$$
$$\dot{S} = -[S - S_T(x,y)]/\tau_S + w_9$$
$$\dot{T} = -[T - T_T(x,y)]/\tau_T + w_{10}$$

where
- w_i represent gaussian white process noise
- f_i resolve the surface velocity into inertial components
- S_T and T_T are angles of the terrain surface as a function of x and y position
- $Z(x,y)$ represents the known surface elevation as a function of x and y position
- g_v is the component of gravity in the direction of the vehicle's velocity vector.

Table 3
Measurement Equations

$$R_m = R + v_1$$

$$S_m = S + v_2$$

$$T_m = T + v_3$$

$$Ra = [x^2 + y^2 + z^2]^{1/2} + v_4$$

$$\dot{R}a = [x\dot{x} + y\dot{y} + z\dot{z}]/[x^2 + y^2 + z^2]^{1/2} + v_5$$

$$\eta = \tan^{-1}\ (y/x) + v_6$$

$$\xi = \tan^{-1}[-z/(x^2 + y^2)] + v_7$$

$$\dot{\eta} = (x\dot{y} - y\dot{x})/(x^2 + y^2) + v_8$$

$$\dot{\xi} = [z\ (x\dot{x} + y\dot{y}) - \dot{z}(x^2 + y^2)]/[(x^2$$

where the v_i represent gaussian white measurement noises

Table 4
Continuous-Discrete Extended Kalman Filter

System Model	$\dot{\underline{x}}(t) = \underline{f}(\underline{x}(t),t) + \underline{w}(t)\ ; \qquad \underline{w}(t) \sim N(\underline{0}, Q(t))$
Measurement Model	$\underline{z}_k = \underline{h}_k(\underline{x}(t_k)) + \underline{v}_k\ ; \qquad k = 1, 2, \ldots\ ; \qquad \underline{v}_k \sim N(\underline{0}, R_k)$
Initial Conditions	$\underline{x}(0) \sim N(\hat{\underline{x}}_o, P_o)$
Other Assumptions	$E\left[\underline{w}(t)\, \underline{v}_k^T\right] = 0$ for all k and all t
State Estimate Propagation	$\dot{\hat{\underline{x}}}(t) = \underline{f}(\hat{\underline{x}}(t),t)$
Error Covariance Propagation	$\dot{P}(t) = F(\hat{\underline{x}}(t),t)\, P(t) + P(t)\, F^T(\hat{\underline{x}}(t),t) + Q(t)$
State Estimate Update	$\hat{\underline{x}}_k(+) = \hat{\underline{x}}_k(-) + K_k[\underline{z}_k - \underline{h}_k(\hat{\underline{x}}_k(-))]$
Error Covariance Update	$P_k(+) = [I - K_k H_k(\hat{\underline{x}}_k(-))]\, P_k(-)$
Gain Matrix	$K_k = P_k(-)\, H_k^T(\hat{\underline{x}}_k(-)) \left[H_k(\hat{\underline{x}}_k(-))\, P_k(-)\, H_k^T(\hat{\underline{x}}_k(-)) + R_k\right]^{-1}$
Definitions	$F(\hat{\underline{x}}(t),t) = \left.\dfrac{\partial \underline{f}(\underline{x}(t),t)}{\partial \underline{x}(t)}\right\vert_{\underline{x}(t)=\hat{\underline{x}}(t)}$ $H_k(\hat{\underline{x}}_k(-)) = \left.\dfrac{\partial \underline{h}_k(\underline{x}(t_k))}{\partial \underline{x}(t_k)}\right\vert_{\underline{x}(t_k)=\hat{\underline{x}}_k(-)}$

9. Figures

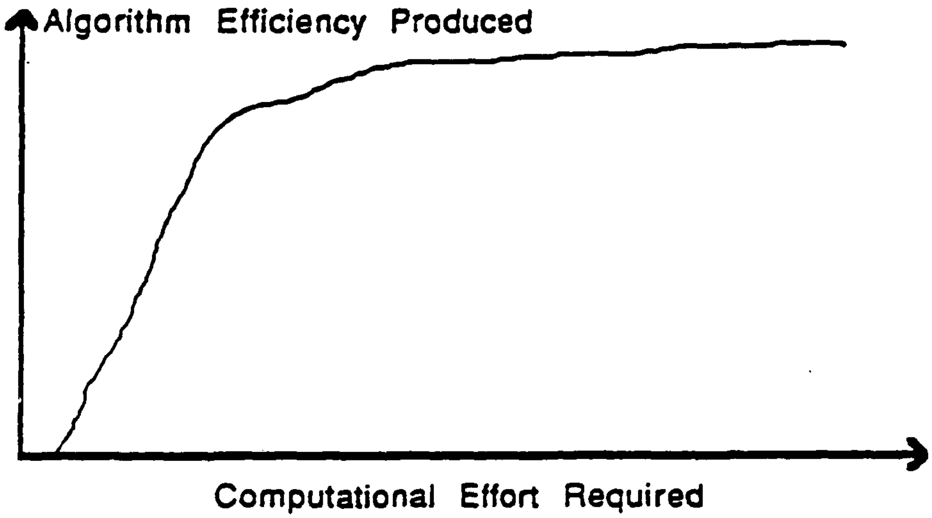


Figure 1: Computational Effort Versus Algorithm Efficacy

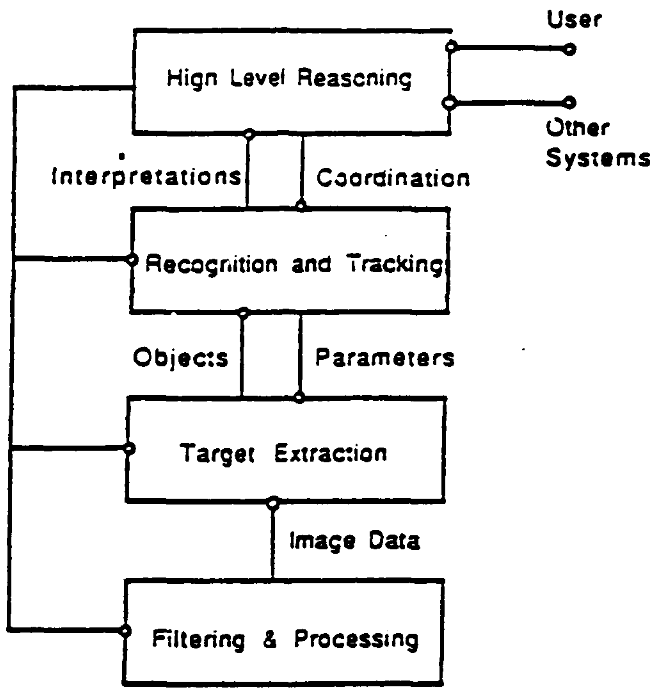


Figure 2: The Hierarchical System Description

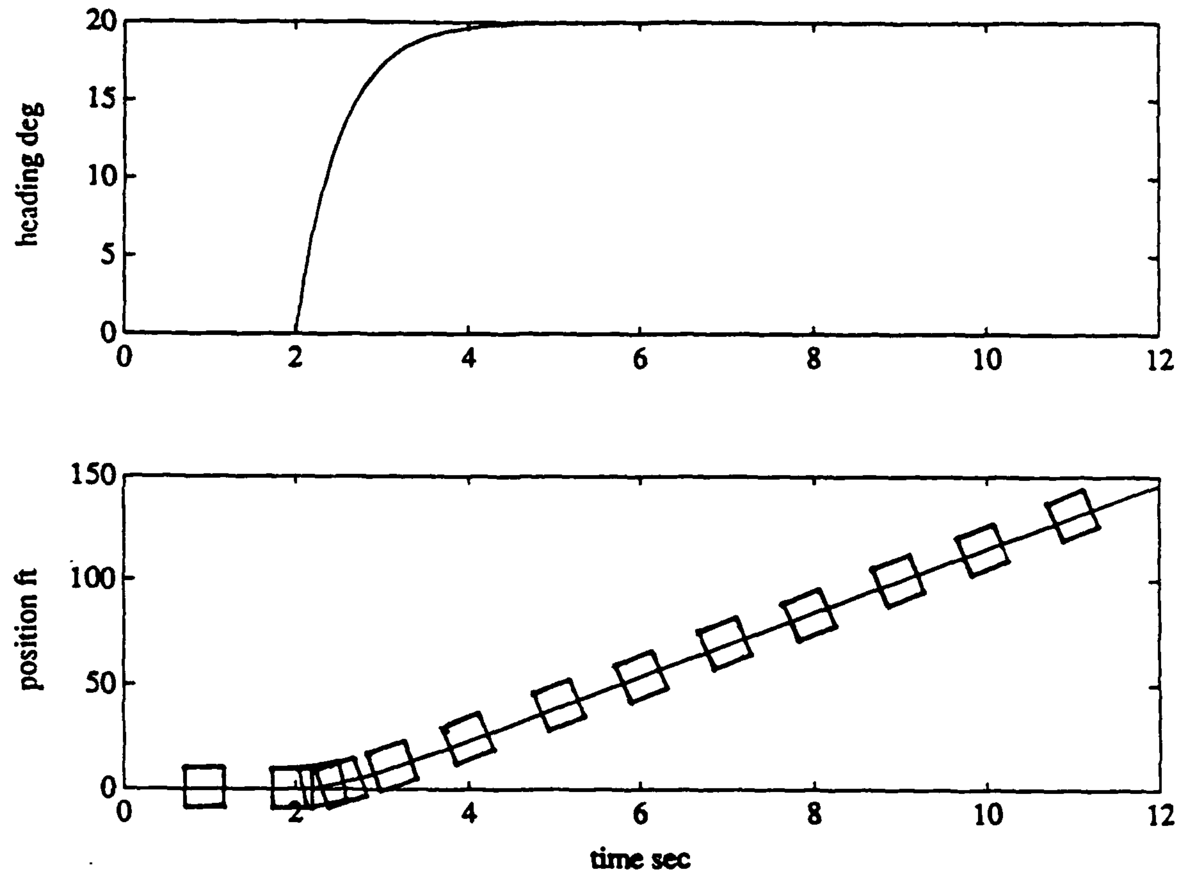


Figure 3: Heading Angle Change Leads Trajectory Change

10. Appendices

Appendix A
A Hierarchical Approach to Passive Robust Target Tracking

By: Andrisani, D., Tenorio, M.F., Lu, J., Kuhl, F.P.

(This paper was presented at the Ninth Meeting of the Coordinating Group on Modern Control, October 24-25, 1989, Picatinny Arsenal, N.J.)

A HIERARCHICAL APPROAGH TO PASSIVE ROBUST TARGET TRACKING

by
Dominick Andrisani, II, M. Fernando Tenorio, Jun Lu
Purdue University, West Lafayette , Indiana
and
Frank P. Kuhl
U.S. Army Armament Research, Development, and Engineering Center
Picatinny Arsenal, Dover, NJ

1. Abstract

Target tracking is approached in this paper using a hierarchical structure of cooperating passive sensors, i.e. optical imaging devices. This structure provides a mechanism for fusing data from different sensors and fusing results from different algorithms. The result is a target tracker that is robust with respect to sensor degradation and algorithm limitations.

The specific algorithm for tracking ground based targets using image data is discussed. In particular the correlation of ground vehicle orientation (from an imaging sensor) with target velocity (of importance to the tracker) is discussed. This correlation makes the tracker described in this paper superior to trackers that don't use image data. Furthermore, the tracker proposed in this paper uses terrain map data to help estimate present target position and to predict future target behavior.

2. Introduction

This paper has two goals. First, it describes a hierarchical target extraction, recognition, and tracking system based on passive sensors that can be completely integrated with other battlefield resources. This first goal serves as an introduction to the balance of this paper and to the companion paper entitled "Resonating Structures for Image Based Tracking" by Tenorio, Andrisani, Codrington and Kuhl[1]. The work described in these papers is being funded by the U.S.Army Armament Research, Development and Engineering Center at Picatinny Arsenal.

The hierarchical system consists of the mutually beneficial interconnection of multiple algorithms operating on several hierarchical levels. Together these algorithms or subsystems can cooperate in the solution of a complex problem beyond the capability of any one algorithm. This framework allows for easy addition, deletion or upgrading of new algorithms as well as communication between algorithms on the same and different levels. Furthermore this hierarchical framework provides extensive information to the users (at the highest level) to allow for performance assessment, human intervention, or operator training. Algorithms on different levels are allowed to deal with the world with different degrees of knowledge abstraction. Algorithms can compute a measure of confidence for each type of sensor output to aid in interpretation of results. Mechanisms for learning both within and between algorithms can be provided. It is felt that this hierarchical structure offers potential for robust operation since the loss of one or more algorithms would result in graceful degradation in the overall system performance.

The second goal of this paper is to describe in greater detail a tracker algorithm that uses target orientation information from an image processor to track ground vehicles. The tracker will be utilized at the mid level of the hierarchical target recognition and tracking system.

3. The Hierarchical Target Extraction, Recognition, and Tracking System

The main philosophy behind hierarchical target extraction, identification and tracking system is that a typical single subsystem (sensor and related algorithms) has a performance versus effort relationship shown in Figure 1. To obtain maximum performance requires a very large effort. However, to obtain near maximum performance requires substantially less effort. Our approach is to design a cooperative system where multiple subsystems work together, and where each subsystem is operating at the "knee" of the performance versus effort curve. This approach recognizes that no one subsystem possesses sufficient performance, but with cooperation among the subsystems desirable overall performance is achievable. Furthermore, overall system robustness and reliability is enhanced while maximizing throughput. Each subsystem is simplified and the overall structure is parallelized for efficient implementation more suitable for real time operation.

In our ongoing research program we are developing the concepts described above, studying the implications of fusing the partial results of each algorithm with cooperating neighbors, and incorporating them in an hierarchical, completely passive, target identification and tracking system, capable of being integrated into a wider battle information network. The system is to be highly adaptive and capable of fusing information from a variety of sources (user-given, world, and sensed) in its decision making. The primary targets being considered are ground vehicles, although airborne targets can also be included. Pure imagery data will be used (possibly multiband), but the incorporation of laser range or radar data is also possible.

The system is composed of four hierarchical levels as shown in Figure 2.

1. Preprocessing Level: The sensing, filtering and distribution subsystems.
2. Low Level: Early processing; responsible for object extraction, tracking using simple image data (e.g. object centroid) and sensor pointing.
3. Mid Level: Secondary processing; responsible for object recognition, object orientation determination, and tracking using object orientation.
4. High Level: Reasoning; responsible for information fusion and coordination, user interaction, maintenance of world knowledge, and "smart" trajectory prediction.

At Purdue University a facility is being developed to implement and test this hierarchical structure. One of the key components in this effort is a battlefield simulation containing the moving targets and the moving tracker including the adjustable cameras and the images they generate. Those images can then be utilized as the primary raw sensor data for input to the hierarchical tracking system. The next sections describe in greater detail the intended implementation of the mid-level ground target tracking algorithm.

4. Justification for Image Based Trackers

On a modern battlefield enemy aircraft and ground vehicles will employ all the capabilities they can muster in order to guarantee their survival and their mission success. These capabilities include speed, maneuverability, and unpredictablity. There will be a substantial advantage to those vehicles that can move stealthfully without emitting radiation to reveal its position or intentions. There will be further advantage to any vehicle that can use the radiation emitted from an enemy vehicle to detect, recognize, track, and distroy that enemy.

These considerations suggest that the advantage will be held by any vehicle that uses passive (nonradiating) means to detect, recognize, and track and distroy its' enemy. To be effective the tracking system must be capable of tracking rapidly moving and maneuvering targets while themselves speeding along.

In this paper we propose a class of hierarchical target tracking systems that are passive and capable of recognizing and tracking maneuvering targets. We employ video based imaging technology as the passive sensor and image processing to detect and track the target. Maneuvers of the target are detected and compensated for by extracting target aspect angle information from the images. This allows us to utilize the relationships that exist between ground vehicle aspect angles and vehicle linear velocity to more accurately determine target velocity. Knowing this we can more accurately predict future target motion, i.e. when firing at the target we can more accurately lead the target[2].

Target aspect angle information is a supplement to the sensor (camera or FLIR) pointing angle information that gives direct data as to the present position of the target. It is also proposed that sensor angular rates be employed to in order to obtain direct data on target velocity.

The achilles heel of image based tracking lies in determing range to the target. We do not claim to have all the answers to this problem. We suggest the following approaches. A pulsed laser rangefinder can be used to determine range to target with great accuracy but at a loss of passive operation. Two separated sensors provide the ability to triangulate on the target or to use stereo imaging techniques to find range. Furthermore, it is also possible to borrow techniques employed by submarines to passively track other vehicles using only the bearing angle information provided by sonar sensors.

5. Application to Ground Vehicles

The key reason that image based target trackers offer the potential for superior performance over radar based trackers is that the image contains important information not included in radar data. That data is the target orientation (e.g. aspect angles or Euler angles).

In the case of fixed wing[3,4] or rotary wing aircraft[5,6] the orientation is directly related to target acceleration. For instance an aircraft turning will be banked into the turn. The amount of acceleration is proportional to the amount of banking. This new information helps the tracker to more accurately estimate target acceleration. This in turn helps the tracker make more accurate predictions of future target motion.

In contrast to aircraft, it is not possible to relate acceleration to orientation for tracked or wheeled vehicles. For example, a ground vehicle turning will generally stay level on the terrain surface even though the tires or tracks are generating side loads to curve the trajectory of the vehicle. Thus, in any single image a vehicle turning looks the same as a vehicle going straight (assuming that the imaging sensor is incapable of seeing the turning wheels).

On the other hand, it is possible to relate ground vehicle orientation to the direction of the velocity vector. This is because ground based vehicles roll over the surface causing the velocity vector to be in the direction that the vehicle is pointed. This is true with tracked vehicles and with wheeled vehicles. While some sliding of the wheels or tracks may frequently occur in battle, the primary mode of locomotion will be from rolling. Furthermore, the fact that rolling friction is greater then sliding friction and the necessity to maintain economy of fuel usage compells most drivers to roll rather then slide their tires or tracks.

In summary, the orientation of the ground vehicle gives direct information as to the direction of the velocity vector. It is believed that a tracker of ground vehicles that uses orientation information will outperform a tracker that does not.

As with aircraft, the change in vehicle orientation (heading angle) will precede a substantial change in cross track position as illustrated in Figure 3. This observation suggests increased accuracy in predicting (leading) the future position of the vehicle.

In this world of increasingly dense computer mass storage it is reasonable to assume that a next generation target tracker will have on-line access to a digital terrain map of the battlefield. This information can be used by the tracker in a number of ways.

1. Since the vehicle being tracked is constrained to move on the surface of the terrain, the terrain map can be used to help the tracker determine his present and furure position. Certainly a vehicle approaching a steep decline can be predicted to loose elevation as he descends down the decline.

2. Since the vehicle will sit flat on the terrain, the orientation of the vehicle will be in part determined by the orientation of the terrain surface. This can be used to help determine the actual orientation of the target (i.e to aid the image processor) and to help track it.

3. The local slope of the terrain may be expected to influence the acceleration of the vehicle, e.g. a vehicle descending a hill will likely speed up.

4. Terrain data is also available with surface conditions, e.g. forests grasslands, marshes, roads. These also can be used by a "smart" tracker to predict future movements.

Tables 1 and 2 describe the state equations of a ground vehicle tracker that utilizes terrain maps in the first three ways mentioned above. Notice that if the tracker has some idea of the commanded speed, commanded heading turn rate, the tracker state equations, when integrated forward in time, will produce a trajectory that keeps the vehicle on the terrain surface at an orientation consistent with the terrain.

Table 3 describes one possible set of measurements that might be used by the tracker. Angles R_m, S_m, and T_m come from the image processor. This set assumes range, Ra, and range rate, $\dot{R}a$, are available to the tracker from a laser rangefinder. This data may not be available if passive operation is required. In that case a second camera might be required to allow triangulation to find range. Angles η and ξ are the camera azimuth and elevation pointing angles respectively. Angular rates $\dot{\eta}$ and $\dot{\xi}$ are optional but useful pointing angular rates of the camera.

Equations of motion given in Table 2 are not designed to accurately simulate the motion of a ground vehicle. Instead, they are designed to track the motion of a ground vehicle given measurements of the motion of the vehicle. In the simulation process, it is important to accurately model all the forces and moments acting on the vehicle. However, in the tracking problem most of those forces and moments are unknown to the tracker. Thus, equations of motion for tracking can be much simpler then those for simulation.

Given the equations of motion and measurement equations as described above, we complete our tracker by utilizing an extended Kalman filter to estimate the present state of the target. In particular we use the "Continuous-discrete extended kalman filter" described in [7] and shown in Table 4. For predicting the future trajectory we integrate the nonlinear equations of motion in Table 2 forward in time assuming no process noise.

At the present time a tracker for ground vehicles of the type described above is being implemented at Purdue University for incorporation into the hierarchical tracking system at the mid level.

6. The Symbiosis Between Image Processing and Tracking

A fortunate benefit of optical target tracking systems is that by tightly linking the image processing subsystem and target tracking subsystem, a symbiotic relationship can exist between these two subsystems thereby improving overall tracker performance.

Specifically, the image processing subsystem enjoys the following benefits from its close association with the target tracking subsystem.

1. Accurate tracking of the maneuvering target allows the optical sensor to employ a longer focal length thereby creating a larger target image within the field of view. In addition, accurate tracking helps eliminate loss of the target from the field of view.

2. Since a target usually is pointed into its velocity vector, the tracker (knowing the velocity vector) can help the image processor resolve the visual ambiguity associated with determining whether the target is pointing into or away from the imaging sensor.

3. The tracker can often provide *a priori* estimates of the location and aspect angle of the target within the sensor's field of view. This can reduce computer time required by many image processors, especially ones that use library search techniques to find target attitude.

The tracking subsystem enjoys the following benefits from its close association with the the image processing subsystem.

1. Location of the target within the image frame (i.e. boresight corrections) can be used to more accurately locate the target on the battlefield. This data supplements the sensor pointing data (i.e. the direction of the sensor boresight).

2. By knowing target aspect angle the tracker can more accurately estimate the direction of the target velocity vector and therefore determine with greater precision the present and future position of the target.

7. Conclusions

Target tracking is approached in this paper using a hierarchical structure of cooperating sensors, i.e. optical imaging devices. This structure provides a mechanism for fusing data from different sensors and fusing results from different algorithms. The result is a target tracker that is robust with respect to sensor degradation and algorithm limitations.

The specific algorithm for tracking ground based targets using image data is discussed. In particular the correlation of ground vehicle orientation (from an imaging sensor) with target velocity (being determined by the tracker) is discussed. This correlation makes the tracker described in this paper superior to trackers that don't use image data. Furthermore, the tracker proposed in this paper uses terrain map data to help estimate present target position and to predict the future target behavior.

8. References

[1] Tenorio, M.F., Andrisani, D., Codrington, C., Kuhl, F.P., "Resonating Structures for Image Based Tracking", presented at the *Ninth Meeting of the Coordinating Group on Modern Control*, October 24-25, 1989, U.S. ARDEC, Picatinny Arsenal, Dover, NJ.

[2] Wrigley, W. and Hovorka, J. *Fire Control Principles*, McGraw-Hill Book Company, Inc. 1959.

[3] Andrisani, D., Kuhl, F.P., Gleason, D., "A Nonlinear Tracker Using Attitude Measurements," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-22, No. 5, September, 1985.

[4] Andrisani, D., Gorman, J.W., Mitchell, O.R., Thurman, S.W., "Orientation Identification and Maneuver Estimation of Tactical Targets Using Video Data," Final Report Battelle Delivery Order No. 0463, August 29, 1983. Prepared for Battelle Columbus Laboratories Scientific Services Program, Research Triangle Park, NC.

[5] Andrisani, D., Kuhl, F.P., Schierman, J.D., "Helicopter Tracking Using Attitude Measurements," *Proceedings of the Eighth Meeting of the Coordinating Group on Modern Control Theory*, Rochester, Michigan, February, 1987.

[6] Andrisani, D., Schierman, J.D., "Tracking Maneuvering Helicopters Using Attitude and Rotor Angle Measurements," *Conference Record: Twenty-First Annual Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, California, November 2-4, 1987, pp. 328-333.

[7] Gelb, A., Editor, *Applied Optimal Estimation* M.I.T. Press, 1974.

Table 1
State Vector for Ground Vehicle Tracker

$$\bar{x} = \begin{bmatrix} V_c & \text{commanded linear velocity along the terrain surface} \\ V & \text{actual linear velocity along the terrain surface} \\ x & \text{North inertial position} \\ y & \text{East inertial position} \\ z & \text{down inertial position} \\ \dot{R}_c & \text{commanded turn rate} \\ \dot{R} & \text{actual turn rate} \\ R & \text{heading angle} \\ S & \text{pitch angle} \\ T & \text{bank angle} \end{bmatrix}$$

Table 2
State Equations for Ground Vehicle Tracker

$$\begin{aligned}
\dot{V}_C &= w_1 \\
\dot{V} &= -(V - V_C)/\tau_V + g_v + w_2 \\
\dot{x} &= V f_1(R,S,T) \\
\dot{y} &= V f_2(R,S,T) \\
\dot{z} &= V f_3(R,S,T) \\
\ddot{R}_c &= w_6 \\
\ddot{R} &= -(\dot{R} - \dot{R}_c)/\tau_R + w_7 \\
\dot{R} &= \dot{R} \\
\dot{S} &= [S - S_T(x,y)]/\tau_S + w_9 \\
\dot{T} &= -[T - T_T(x,y)]/\tau_T + w_{10}
\end{aligned}$$

where
- w_i represent gaussian white process noise
- f_i resolve the surface velocity into inertial components
- S_T and T_T are angles of the terrain surface as a function of x and y position
- g_v is the component of gravity in the direction of the vehicle's velocity vector.

Table 3
Measurement Equations

$$R_m = R + v_1$$

$$S_m = S + v_2$$

$$T_m = T + v_3$$

$$Ra = [x^2 + y^2 + z^2]^{1/2} + v_4$$

$$\dot{Ra} = [x\dot{x} + y\dot{y} + z\dot{z}]/[x^2 + y^2 + z^2]^{1/2} + v_5$$

$$\eta = \tan^{-1}(y/x) + v_6$$

$$\xi = \tan^{-1}[-z/(x^2 + y^2)] + v_7$$

$$\dot{\eta} = (x\dot{y} - y\dot{x})/(x^2 + y^2) + v_8$$

$$\dot{\xi} = [z(x\dot{x} + y\dot{y}) - \dot{z}(x^2 + y^2)]/[(x^2 + y^2 + z^2)(x^2 + y^2)^{1/2}] + v_9$$

where the v_i represent gaussian white measurement noises

Table 4
Continuous-Discrete Extended Kalman Filter

System Model	$\dot{\underline{x}}(t) = \underline{f}(\underline{x}(t),t) + \underline{w}(t)$; $\underline{w}(t) \sim N(\underline{0}, Q(t))$
Measurement Model	$\underline{z}_k = \underline{h}_k(\underline{x}(t_k)) + \underline{v}_k$; $k = 1, 2, \ldots$; $\underline{v}_k \sim N(\underline{0}, R_k)$
Initial Conditions	$\underline{x}(0) \sim N(\hat{\underline{x}}_o, P_o)$
Other Assumptions	$E\left[\underline{w}(t)\,\underline{v}_k^T\right] = 0$ for all k and all t
State Estimate Propagation	$\dot{\hat{\underline{x}}}(t) = \underline{f}(\hat{\underline{x}}(t),t)$
Error Covariance Propagation	$\dot{P}(t) = F(\hat{\underline{x}}(t),t)\,P(t) + P(t)\,F^T(\hat{\underline{x}}(t),t) + Q(t)$
State Estimate Update	$\hat{\underline{x}}_k(+) = \hat{\underline{x}}_k(-) + K_k[\underline{z}_k - \underline{h}_k(\hat{\underline{x}}_k(-))]$
Error Covariance Update	$P_k(+) = [I - K_k H_k(\hat{\underline{x}}_k(-))]\,P_k(-)$
Gain Matrix	$K_k = P_k(-)\,H_k^T(\hat{\underline{x}}_k(-))\left[H_k(\hat{\underline{x}}_k(-))\,P_k(-)\,H_k^T(\hat{\underline{x}}_k(-)) + R_k\right]^{-1}$
Definitions	$F(\hat{\underline{x}}(t),t) = \dfrac{\partial \underline{f}(\underline{x}(t),t)}{\partial \underline{x}(t)}\bigg\rvert_{\underline{x}(t)=\hat{\underline{x}}(t)}$ $H_k(\hat{\underline{x}}_k(-)) = \dfrac{\partial \underline{h}_k(\underline{x}(t_k))}{\partial \underline{x}(t_k)}\bigg\rvert_{\underline{x}(t_k)=\hat{\underline{x}}_k(-)}$

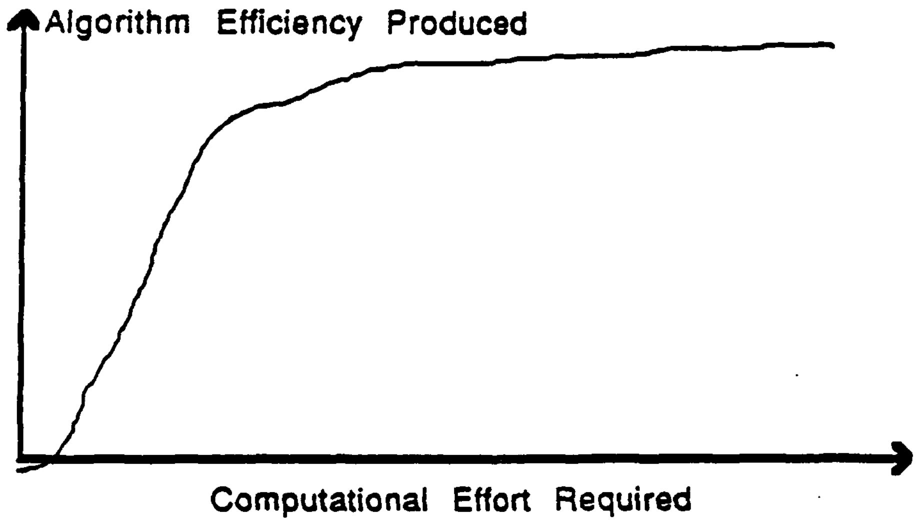


Figure 1: Computational Effort Versus Algorithm Efficacy

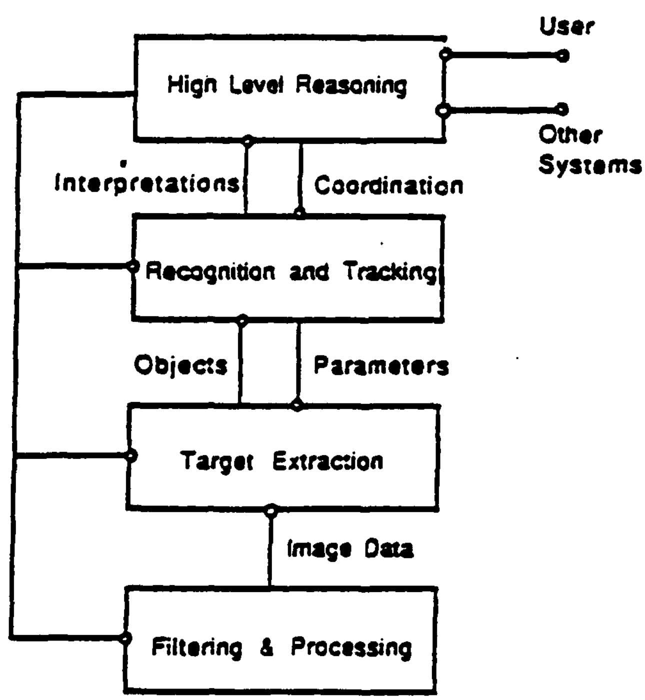


Figure 2: The Hierarchical System Description

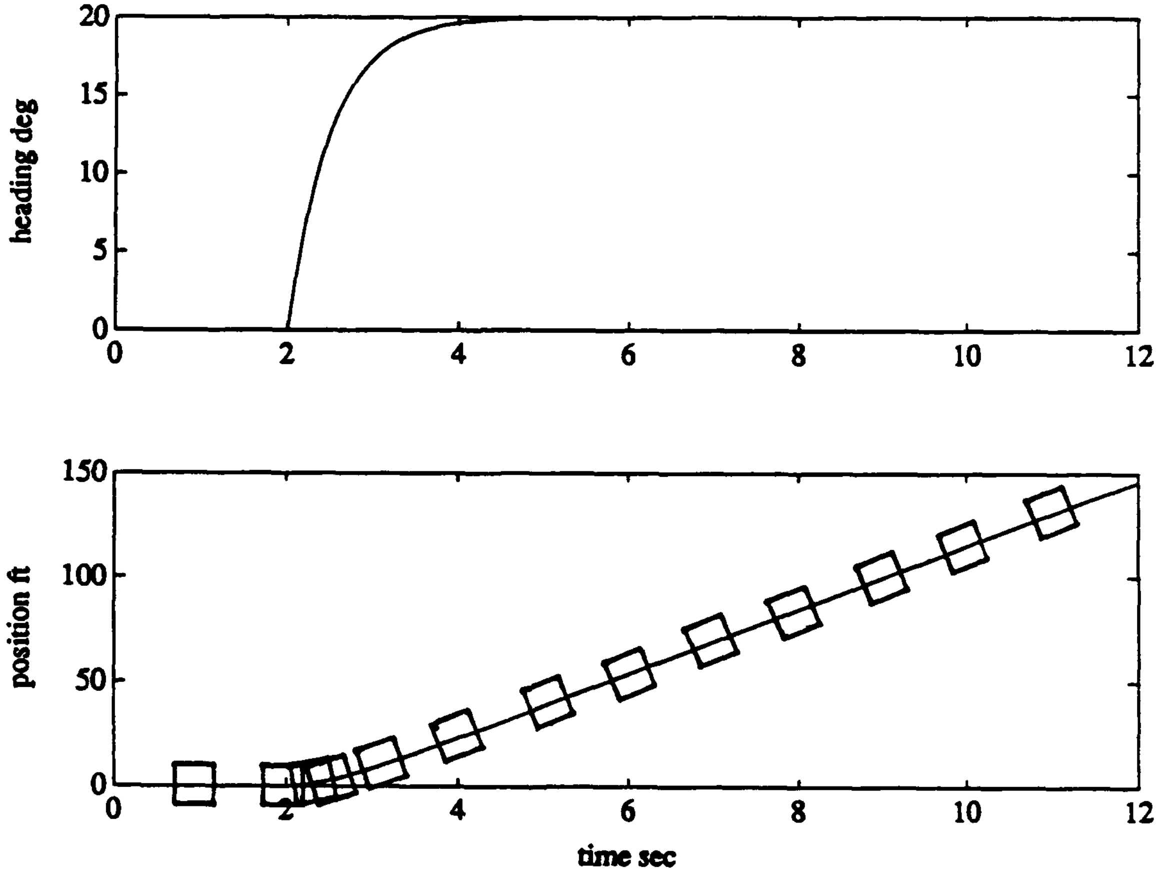


Figure 3: Heading Angle Change Leads Trajectory Change

Appendix B
Resonating Structures for Image Based Tracking

By: Tenorio, M.F., Andrisani, D., Codrington, C., Kuhl, F.P.

(This paper was presented at the Ninth Meeting of the Coordinating Group on Modern Control, October 24-25, 1989, Picatinny Arsenal, N.J.)

Abstract

The design of classical vision systems is based on serially piecing together individual algorithms, each of which is intended to solve a specific part of the vision problem under a given set of assumptions. This has met with poor performance and low information throughput. The design process for each of these algorithms is usually disjoint, and ignores the system integration process. Here, we speculate on a new philosophy for the design of vision systems that uses highly parallel and simple elements that are easily integrated. To improve the performance of the simple algorithms involved, heavy use of closed feedback loops is made throughout the system. These loops have self correcting capabilities in different time scales. This work suggests a simple system that uses several of these concepts object recognition in noisy conditions.

Appendix C
The Design of an Insensitive Estimator for Target Tracking

By: Andrisani, D., Kim, E.T.

(This paper has been submitted for presentation at the 1990 AIAA Guidance and Control Conference, August 1990, Portland, Oregon)

THE DESIGN OF AN INSENSITIVE ESTIMATOR FOR TARGET TRACKING

Dominick Andrisani,II and Eung T. Kim
School of Aeronautics and Astronautics
Purdue University
West Lafayette, IN 47907

Abstract

Two related methods for designing estimators to be insensitive to unknown variations in parameters of the plant are presented. A mechanism for trading off small estimator error variance and low sensitivity to unknown parameter variations has been developed using estimator gains computed from an iterative algorithm. This insensitive estimator design method is applied to the target tracking problem. In this example an α - β tracking filter is designed to be insensitive to the type af aircraft being tracked (i.e. the mathematical model of the aircraft being tracked).

1. Introduction

When tracking an unfriendly target aircraft the mathematical model of the target, i.e the order and the nature of its parameterization, is usually unknown. Furthermore the parameters of that mathematical model, such as stability and control derivatives, would generally be unknown and time varying. To make matters worse, the often violent control inputs applied by the pilot are also unknown to the tracker. Designing a successful state estimator for a target tracker under these conditions is indeed a difficult task. An adaptive estimator can improve the performance in the above situation by using the information acquired during the operation of the estimator to change some internal property of the estimator. However the adaptation mechanism often places an excessive computational load on a real time tracking algorithm. The approach taken in this paper is to design a nonadaptive estimator that is insensitive to parameter variations.

Estimators insensitive to parameter variation have not received as much attention in the literature as insensitive controllers. Most of the studies about robust estimation address the situation where measurement disturbances are not Gaussian. F. J. Alexando suggested a computationally simple method by making symmetrical the system matrix of the estimator, $(A - KC)$, in order to design observers insensitive to parameter changes in the presence of slowly varying inputs and no measurement and process noise [1].

In this paper a new design method for estimators insensitive to parameter variations is developed by modifying the cost function used in the design of insensitive controllers. The performance of this insensitive estimator is illustrated using in the context of target tracking.

2. Background: Design of Insensitive Estimators

The continuous, linear, time-invariant plant dynamics for a target aircraft can be described by the linear differential equations

$$\dot{x} = A(a)x + B(a)w \tag{1}$$

where a is a vector of system parameters and w is a zero-mean white Gaussian process noise vector with spectral density Q. The system input is modeled here using process noise to approximate the unknown inputs applied to the aircraft by the pilot and by atmospheric turbulence. The measurement output equation is given by

$$y = C(a)x + v \tag{2}$$

where v is a zero-mean white measurement noise vector with spectral density R.

It is very common in the design of estimators (as in the target tracking case) that the exact plant matrices A(a), B(a) and C(a) are unknown to the designer of the estimator. Instead the designer of the estimator must frequently use a different perhaps simpler state space model. We assume here that the designer of the estimator has chosen to use A_o, B_o and C_o system matrices to design his state estimator. His design parameters are then the estimator gains K. In selecting K, he is concerned about the sensitivity of his estimator to variations in the plant matrices A(a), B(a) and C(a).

We assume the estimator to be of the following form

$$\dot{\hat{x}} = A_0 \hat{x} + K (y - C_0 \hat{x}) \qquad (3)$$

where $\hat{x}$ = estimated state vector

K = gain of the estimator

A_0 = estimator system matrix (not necessarily same as $A(a)$)

C_0 = estimator output matrix (not necessarily same as $C(a)$)

The gains K can be determined by numerous procedures. If the Kalman Filter is used, the assumption must be made that $A_0 = A(a_0)$ and $C_0 = C(a_0)$ and that a_0 is the correct parameter vector. The Kalman Filter then minimizes the following cost function.

$$J(a_0, K) = \lim_{t \to \infty} [E (e^T W e)]_{a = a_0} \qquad (4)$$

where e is estimation error (*i.e.* $e \triangleq x - \hat{x}$), W is a weighting matrix and a_0 is a nominal value of the uncertain parameters.

The Kalman Filter is only optimal when the design conditions are exactly true and exhibits degraded performance whenever parameter vector a varies from those values used in the filter design (a_0). Actual estimator performance may be highly sensitive to parameter variation.

As a way of dealing with parameter sensitivity, "the expected cost method" was developed by Ly and Cannon [2] who used the expected value of the cost over the complete range of parameter variations as a cost function. Since the calculation of the expected value of the cost requires considerable computation for a continuous probability density function, the idea of using a finite sum of costs, each one evaluated at a fairly small number of points in the range of parameter variation was considered [3]. This method of minimizing the sum of cost will be called a "minisum" method here for convenience.

The cost function used in the "minisum" controller design[4] may also be used for estimator design and be written as follows

$$J(K) = \sum_{i=1}^{k} P_i J_i(a_i, K) \tag{5}$$

where P_i is the probability of $a = a_i$ and

$$J_i(a_i, K) = \lim_{t \to \infty} [\, E\,(e^T W e\,)]_{a = a_i} \tag{6}$$

With the "minisum" design procedure, estimator gains K would be chosen to minimize J(K) in Eq.(5).

The estimator designed using the above procedure would be insensitive in some sense. But insensitiveness can be defined in several ways. The "minimax" method is another popular insensitive design method [5]. With the "minimax" design procedure estimator gains K would be chosen to minimize the largest $P_i J_i(a_i, K)$. This approach requires the determination of the parameter giving worst performance. This can be quite expensive computationally [3]. To compare the inherent properties of two methods - "minimax" and "minisum", an example is shown in Fig.1 where the values of performance indices weighted by probability are drawn against the plant parameter variation.

In Fig.1, let's assume (a) was obtained by the "minisum" method and (b) is a "minimax" design result. It is clear in this case that the "minisum" method result is highly sensitive to parameter variation and "minimax" design will be preferred even though it has a little bit larger expected cost. But if (c) is assumed to be obtained from the "minimax" method instead of (b), the "minimax" estimator is not as good since its insensitiveness is meaningless because of its large expected cost. The choice of design method between these two cannot be determined until performance indices are calculated as a function of system parameters.

A new cost criterion which could avoid the above problem can be obtained by a simple modification of the cost function defined by Eq.(5). The idea is to give more weight to large contributors in the performance index and less weight to smaller ones. This can be done by choosing $(P_iJ_i)^{m-1}$ as a weight function, since this $(P_iJ_i)^{m-1}$ is large when P_iJ_i is large and small when P_iJ_i is small for all $m > 1$.

The new cost criterion can be written as follows

$$J_A(K) = \sum_{i=1}^{k} [\, P_iJ_i(a_i, K)\,]^m \tag{7}$$

Now the objective is to find estimator gains, K which minimize $J_A(K)$ with properly chosen value of m under the assumption A_o, B_o, and C_o are fixed. Note that when $m = 1$, this method is exactly same as "minisum" method, and as $m \rightarrow \infty$, the method approaches to the "minimax" method. (Too large value of m cannot be used because of numerical problems.) This new method will be called the "asymptotic minimax" method here.

3. Design Method #1: The Insensitive Kalman Filter

The Kalman Filter uses gain obtained by solving the Riccati equation.

$$0 = SA_o^T + A_o S - SC_o^T R^{-1} C_o S + B_o Q B_o^T$$
$$K = SC_o^T R^{-1}$$

The Kalman gain is a function of both Q and R. Measurement noise spectral density R may be assumed to be known to the designer, since the specifications of measurement devices are available to him. But process noise spectral density Q of the plant is not generally known, and Q loses its meaning as the spectral density of the process noise in the practical situation where A_o, B_o and C_o matrices are not same as A, B and C respectively. This naturally leaves Q to be used as a design parameter.

In order to make the Kalman Filter insensitive to parameter variations, Q can be adjusted to minimize $J_A(K)$. In doing this the gain is constrained to follows the Riccati equation given above. In many cases, a line search algorithm such as the Golden Section Method [6] can be used to find this value of Q (and therefore the Kalman gain K).

An estimator designed with the procedure described above will be called the "Insensitive Kalman Filter". It should be noted that the gains computed this way are not necessarily the gains that minimize $J_A(K)$. Gains found without the Riccati equation constraint will probably produce a smaller cost.

4. Design Method #2: The Asymptotic Minimax Method

In this section we present a design procedure that does not use the Riccati equation constraint of the Insensitive Kalman Filter. Rather the gains are found which directly minimize $J_A(K)$. The method we will call the "asymptotic minimax" method.

The Dynamic equations (1), (2) and (3) can be combined as

$$\begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} = \begin{bmatrix} A & 0 \\ KC & A_o - KC_o \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix} + \begin{bmatrix} B & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} \tag{8}$$

or

$$\dot{x}_a = A_a x_a + B_a u_a \tag{9}$$

If hx is the portion of the state vector to be estimated and $h_o\hat{x}$ is the corresponding portion of the estimated state vector, then the error vector and error variance matrix can be given as

$$e = hx - h_o\hat{x}$$

$$\lim_{t\to\infty} E(eWe^T) = \mathrm{trace}(X\,N) \tag{10}$$

where

$$X \triangleq E\begin{bmatrix} xx^T & x\hat{x}^T \\ \hat{x}x^T & \hat{x}\hat{x}^T \end{bmatrix} \quad N \triangleq \begin{bmatrix} h^T W h & -h^T W h_o \\ -h_o^T W h & h_o^T W h_o \end{bmatrix}$$

The steady state value of X is given by the solution of the Lyapunov equation

$$A_a X + X A_a^T + B_a U_a B_a^T = 0 \tag{11}$$

where

$$U_a = \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix}$$

The existence of unique solution to Eq.(11) is guaranteed if plant system and estimator are stable [7]. The cost function defined in Eq.(7) can be written as

$$J_A(K) = \sum_{i=1}^{k} [P_i \text{ trace } (X_i N_i)]^m \tag{12}$$

where subscript i denotes the evaluation for parameter i. The objective is now to find K minimizing $J_A(K)$ under the constraint of Eq.(11). The first order necessary condition for minimization is that

$$\frac{dJ_A}{dK} = 0 \tag{13}$$

The gradient of $J_A(K)$ can be calculated by introducing a Lagrange multiplier matrix, G, which satisfies the following Lyapunov equation.

$$A_a^T G + G A_a + \frac{\partial [P_i \text{ trace } (X_i N_i)]^m}{\partial X} = 0 \tag{14}$$

We can express G and X in 2 x 2 block matrices as follows

$$G = \begin{bmatrix} G_{11} & G_{12} \\ G_{12}^T & G_{22} \end{bmatrix} \qquad X = \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^T & X_{22} \end{bmatrix}$$

Then the gradient of $J_A(K)$ is given by

$$\frac{\partial J_A}{\partial K} = 2\sum_{i=1}^{k}[\, G_{12_i}{}^T(X_{11_i}C_i{}^T - X_{12_i}C_o^T) + G_{22_i}(X_{12_i}{}^T C_i^T - X_{12_i}C_o^T)$$

$$+ G_{22_i}KR_i] \qquad (15)$$

where $X_{11_i}, X_{12_i}, X_{12_i}, G_{12_i}$ and G_{22_i} are calculated by the following equations.

$$A_iX_{11_i} + X_{11_i}A_i^T + B_iQ_iB_i^T = 0 \qquad (16)$$

$$A_iX_{12_i} + X_{12_i}(A_o - KC_o)_T + X_{11_i}C_i^TK^T = 0 \qquad (17)$$

$$(A_o-KC_o)X_{22_i}+X_{22_i}(A_o-KC_o)^T+KC_iX_{12_i}+(KC_iX_{12_i})^T+KR_iK^T= 0 \qquad (18)$$

$$(A_o - KC_o)^TG_{22_i} + G_{22_i}(A_o - KC_o) + M_{22_i}{}^T = 0 \qquad (19)$$

$$A_i^TG_{12_i} + G_{12_i}(A_o - KC_o) + (KC_i)^TG_{22_i} + M_{12_i}{}^T = 0 \qquad (20)$$

where

$$M_i \triangleq \begin{bmatrix} M_{11_i} & M_{12_i} \\ M_{12_i}{}^T & M_{22_i} \end{bmatrix} \triangleq m[\, P_i\mathrm{trace}(X_iN_i)]^{m-1}$$

Since Eqs.(15-20) are coupled, estimator gain K must be solved by a iteration. A substitution method for the new K from Eq.(15) or a gradient technique such as the DFP algorithm [8] can be used to determine K minimizing J_A of (12). Any Kalman filter gain can be used as an initial value for the iteration since this guarantees a stable estimator.

At each iteration, the new value of K must stabilize the estimator. A minimum of $J_A(K)$ at each iteration will occur before a stability boundary is reached since $J_A(K)$ becomes extremely large as K approaches the stability boundary of estimator. Therefore, the estimator will remain stable by using a sufficiently small increment of K for a new K [9].

5. Application to Aircraft Tracking

In this section we apply the insensitive estimator design methods described above to the problem of tracking the vertical motion of an aircraft. Only estimation was considered. State prediction was considered. The target equations of motion and measurement equation are represented by Eq.(1) and (2) where

$$x = [\, u\ w\ q\ \theta\ z \,]^T$$

$$A = \begin{bmatrix} T_u \cos i_t + X_u + & X_w + X_{\dot{w}} \dfrac{Z_w}{1 - Z_{\dot{w}}} & X_q + X_{\dot{w}} \dfrac{Z_q + U}{1 - Z_{\dot{w}}} & -g\cos\gamma_o - & 0 \\ \dfrac{-T_u \sin i_t + Z_u}{1 - Z_{\dot{w}}} & \dfrac{Z_w}{1 - Z_{\dot{w}}} & \dfrac{Z_q + U}{1 - Z_{\dot{w}}} & -\dfrac{g\sin\gamma_o}{1 - Z_{\dot{w}}} & 0 \\ \dfrac{z_{jm}}{I_{yy}} T_u + M_u + M_{\dot{w}} \dfrac{-T_u \sin i_t + Z_u}{1 - Z_{\dot{w}}} & M_w + M_{\dot{w}} \dfrac{Z_w}{1 - Z_{\dot{w}}} & M_q + M_{\dot{w}} \dfrac{Z_q + U}{1 - Z_{\dot{w}}} & -M_{\dot{w}} \dfrac{g\sin\gamma_o}{1 - Z_{\dot{w}}} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -U & 0 \end{bmatrix} \tag{21}$$

$$B = \begin{bmatrix} X_{\delta_e} + X_{\dot{w}} \dfrac{Z_{\delta_e}}{1 - Z_{\dot{w}}} \\ \dfrac{Z_{\delta_e}}{1 - Z_{\dot{w}}} \\ M_{\delta_e} + M_{\dot{w}} \dfrac{Z_{\delta_e}}{1 - Z_{\dot{w}}} \\ 0 \\ 0 \end{bmatrix} \tag{22}$$

$$C = [\,0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0\,] \tag{23}$$

For the definition of the variables in A and B matrices, refer to reference [10]. The system input, elevator angle, is modeled as white Gaussian noise [11]. The available measurement is assumed to be vertical position z.

As an estimator model, the $\alpha-\beta$ filter was used. This is a typical modern tracking filter with only two states, position and velocity of the target. Normally three $\alpha-\beta$ filters are used in tracking the aircraft, one in each direction of an inertial rectangular coordinate system. Each estimator is described as follows [13]

$$\dot{x} = w$$

where x denotes position in either north, east, or vertical directions while w denotes white noise. Only one $\alpha-\beta$ filter is needed here since target aircraft was assumed to be in vertical motion only. For the $\alpha-\beta$ filter, estimated state vector $\hat{x}$, A_o, B_o and C_o matrices in Eq.(3) are defined as follows.

$$\hat{x} = [\, z \;\; \dot{z}\,]^T$$

$$A_o = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \quad B_o = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad C_o = [1 \quad 0]$$

The flight data with 26 different flight conditions for several types of aircraft were obtained from reference [14,15,16] and listed in Table 1. In the definition of the cost criterion in Eq.(7), subscript i denotes the evaluation at the i th flight condition and k=26 for this example and probability that tracker could encounter with one aircraft in Table 1 was assumed to be same as other aircrafts, that is P_i = 1/26. Pilot input to each aircraft was assumed to be zero-mean white noise whose intensity was determined such that variance of the acceleration in Z-direction of each aircraft is same $(2G)^2$ as for all flight conditions.

The error variances of the estimator were calculated analytically by solving the Lyapunov equations. Fig.2 shows the error variances of the estimators designed by several ways. (a) is the result of Kalman filter whose gain was calculated from Riccati equation by finding the best value of Q minimizing the cost criterion defined in Eq.(7) by line search algorithm. (b), (c) and (d) are the results of "asymptotic minimax" method with m =1, 10 and 20 respectively. And (e) is the "minimax" estimator result, this was obtained by the Standard Kalman filter designed at the worst flight condition.

Note that the sum of error variances of (a) is larger than that of (b). Recall that the gains for (a) are constrained to satisfy a Riccati equation while the gains for (b), (c) and (d) are not so constrained.

Note also that, as m increases, maximum of error covariances decreases, while sum of error variances increases, approaching to "minimax" design result of (e).

By increasing the value of m from 10 to 20, only a slight improvement was achieved in the worst condition at the expense of large performance degradation in other conditions and "minimax" design, (e) is certainly not a good one compared to (d) for this reason. "Asymptotic minimax" method with m=10 is proposed as the most favorable design method in this tracking problem if low sensitivity and small error variance are both required.

As a next example, sensitivity to the change of one stability derivative of aircraft was investigated. The trajectory of T-38 aircraft with flight condition 5 in Table 1 was estimated using an $\alpha-\beta$ estimator. C_{m_α} of the aircraft was changed from -1.3 to -0.1 which is the typical range of C_{m_α} of target vehicles. C_{m_α} is linearly related to M_w in equation (21), see [10]. Seven points of C_{m_α} in this range were considered (k=7) and the probability, P_i was assumed to be same for all cases of C_{m_α}. The principle property of "asymptotic minimax" estimator with different values of m is more clearly shown in Fig.3, and "asymptotic minimax" method with m =10 is felt as the best one again.

5. Conclusions

Two iterative algorithms for computing the state estimator gains was presented with which a trade-off between error variance and sensitivity to parameter variation can be achieved. These methods were developed from the insensitive controller design method suggested by Ly and Cannon, but are much more flexible since they give a free design parameter, m to the designer. If m=1, they become the "minisum" estimator and as $m \rightarrow \infty$, they approache to "minimax" method. A compromise between "minimax" and "minisum" estimator can be obtained by an appropriate choice of m that doesn't have the inherent disadvantages of either the "minimax" and "minisum" estimators. A target tracking numericl example is presented to illustrate the performance of new design methods.

In the design algorithms developed here, it was assumed that estimator model has fixed parameters. However this assumption is not necessary. The optimal values for these parameters can be computed by computing the partial derivatives of the cost function with respect to these parameters. In addition the same design concept used here could be used to design a class of insensitive controllers which would have same properties stated above.

References

1. Alexando, F.J., "Parameter Insensitive Observers," *Proceedings of the Conference on Decision and Control,* 1986, pp1645-1646.

2. Ly, U. and Cannon, R.H., Jr., "A Direct Method for Designing Robust Optimal Control Systems," *Proc. AIAA Guidance Control Conference,* Palo Alto, CA, August 1978, pp.440-448.

3. Vinkler, A. and Wood L.J., "A Comparison of Several Techniques for Designing Controllers of Uncertain Dynamic Systems," *Proceedings of the Conference on Decision and Control,* 1978, pp31-38.

4. Ashkenazi, A. and Bryson, A.E., Jr., "Control Logic of Parameter Insensitivity and Disturbance Attenuation," *Journal of Guidance and Control,* Vol. 4, July-August 1982, pp383-388.

5. Salmon, D.M., "Minimax Controller Design," *IEEE Trans. on Automatic Control,* Vol. AC-13, No.4, August 1968.

6. Luenberger, David G. *Linear and Nonlinear programming* Addison-Wesley, 1984.

7. Kwakernaak, B.D.O. and Moore, J.B., *Linear Optimal Control,* Wiley Interscience, New York, 1972.

8. Horisberger, H.P. and Belanger, P.R. "Solution of the Optimal Constant Output Feedback Problem by Conjugate Gradients," *IEEE Trans. on*

Automatic Control, Vol. AC-19, pp434-435, August 1974.

9. Choi. S.S. and Sirisena, "Computation of Optimal Output Feedback Gains for Linear Multivariable Systems," *IEEE Trans. on Automatic Control,* Vol. AC-19, pp257-258, June 1974.

10. Smetana, F.O., *Computer Assisted Analysis of Aircraft Performance Stability And Control,* McGraw-Hill Book Co., 1984.

11. Singer, R.A., "Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets," *IEEE Trans. on Aerospace and Electronic Systems,* Vol. Aes-6, No.4, July 1970.

12. Moose, R.L. and Gholson N.H., "Adaptive Tracking of Abruptly Maneuvering Targets," *Proceedings of the Conference on Decision and Control,* 1976, pp804-808.

13. Fitzgerald, R.J. "Simple Tracking Filters: Closed Form Solutions," *IEEE Trans. on Aerospace and Electronic Systems,* Vol. Aes-17, No.6, November 1981.

14. McRuer, D., Ashkenas, I. and Graham, D., *Aircraft Dynamics and Automatic Control,* Princeton University Press, 1973.

15. Heffley, R.K. and Jewell, W.F., *Aircraft Handling Qualities Data,* NASA CR 2144, December 1972.

16. Jones, B.W. and Schmidt, D.K., *Flight Simulator - A&AE Minicomputer System*, A&AE Departmental Document #78-53, School of Aeronautics and Astronautics, Purdue University, August 1978.

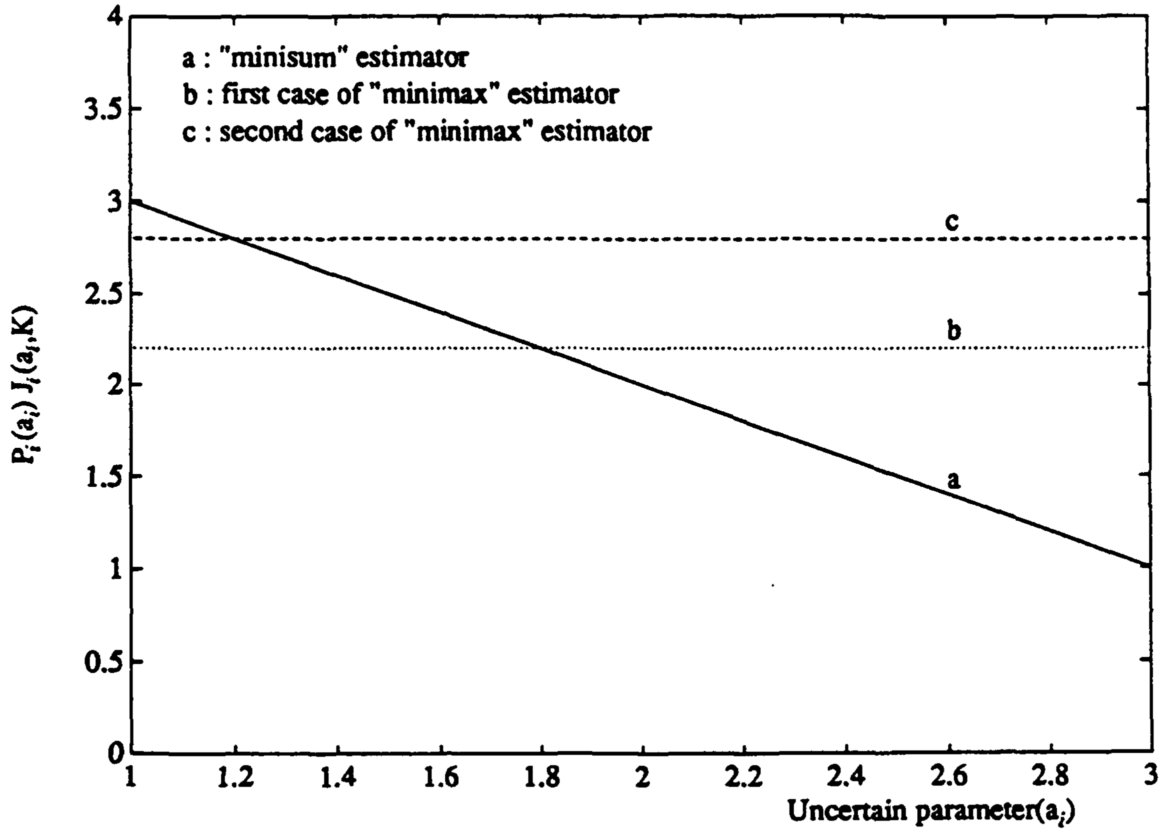


Fig.1 Extreme examples of "minmax" and "minisum" estimator

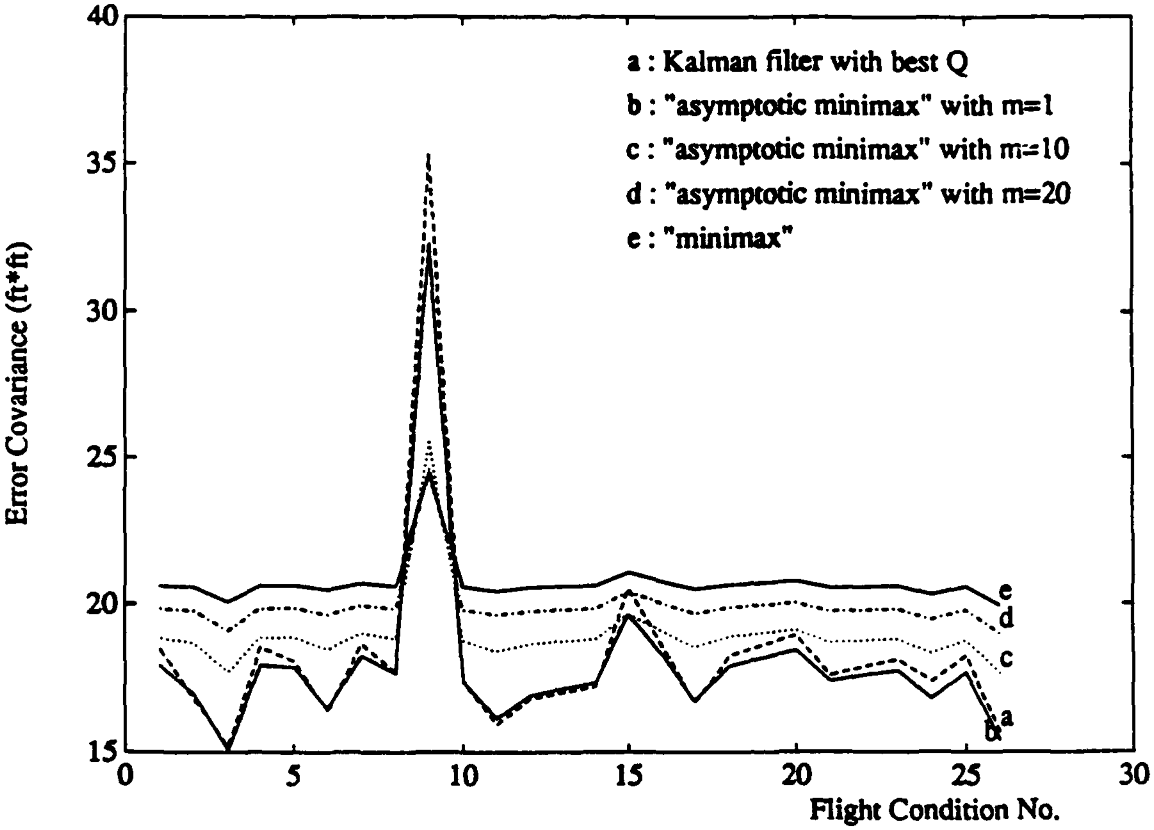


Fig.2 Alpha-beta estimator for 26 different flight conditions

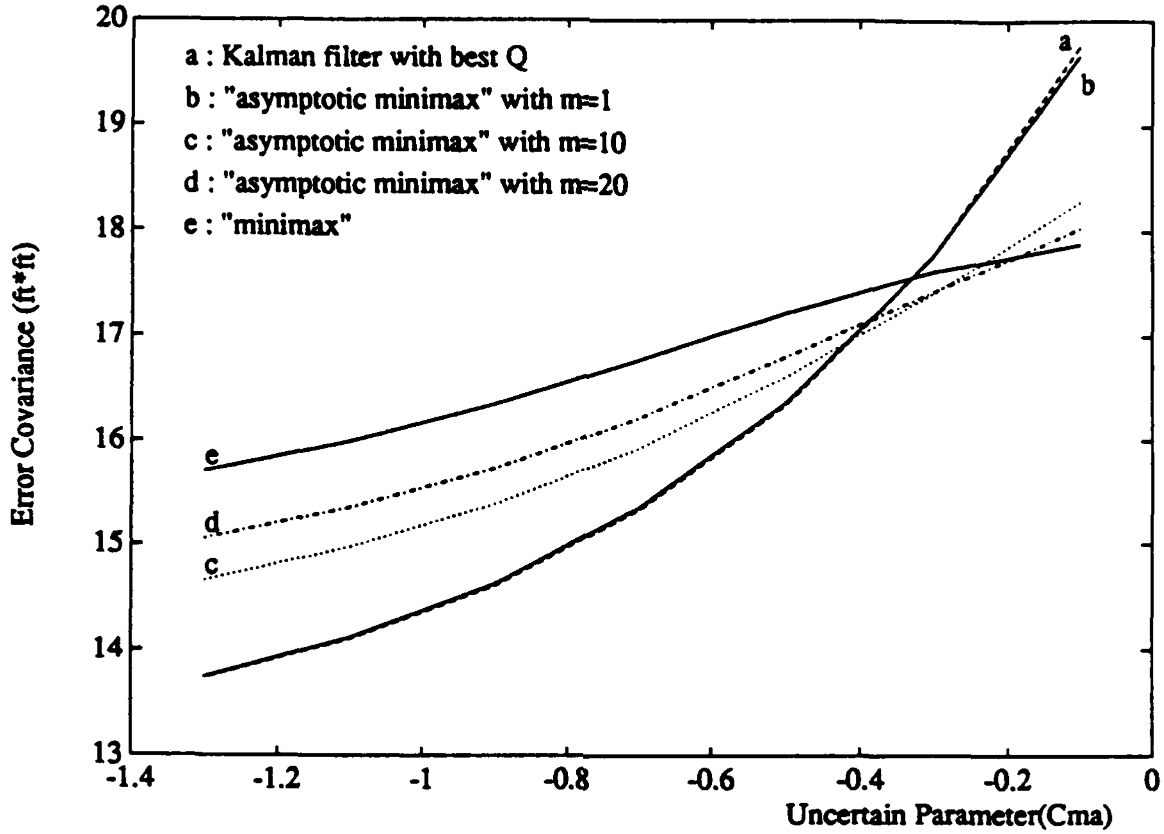


Fig.3 Alpha-beta estimator for Cma variation of T-38

Table.1 Target Aircraft Flight Condition

flight cond. No.	Aircraft	Velocity (ft/sec)	Altitude (Kft)	Mass (slug)
1	F-104A	287	0	14126
2	F-104A	893	0	16300
3	F-104A	1228	0	16300
4	A-4D	223	0	22058
5	A-4D	447	0	17578
6	A-4D	950	0	17578
7	A-4D	423	15	17578
8	A-4D	634	15	17578
9	F-4C	230	0	33197
10	F-4C	893	0	38925
11	F-4C	1228	0	38925
12	T-38	829	15	7500
13	T-38	829	15	9375
14	T-38	829	15	11250
15	NT-33A	228	0	11800
16	NT-33A	447	0	13700
17	NT-33A	782	0	13700
18	B-52	627	10	325000
19	B-52	627	10	406000
20	B-52	627	10	488000
21	B-1	951	0	227950
22	B-1	951	0	313475
23	B-1	951	0	395000
24	C-5A	246	0	580756
25	C-5A	502	0	654399
26	C-5A	726	0	654399

Appendix D
High Level Reasoning Module

Author: Edson Baptista

1. Overall Project Objective

This research is aimed at studying a hierarchical target extraction, identification and tracking system based on passive sensors, that could be completely integrated with other battlefield resources. This research has two major subgoals. The first subgoal is to study and formulate a hierarchical system description for the mutually beneficial interconnection of multiple algorithms operating on several hierarchical levels. Together these algorithms would cooperate in the solution of a complex problem beyond the capability of any algorithm. The second subgoal is to illustrate this hierarchical system description by considering the target extraction, recognition and tracking problem. It is proposed do study cooperating algorithms for a robust passive tracking system capable of accurately determining the present and future trajectories of ground and airborne vehicles.

2. Proposed Solution

A hierarchical structure such as figure-1 of section 5 of this appendix, is the proposed solution.

As can be seen from the above mentioned figure, the solution of such complex problem would be impossible to be executed by a single algorithm, but rather by the integration of several algorithms in several levels of definition of the problem. Furthermore, with this type of solution we can allow that more than one algorithm could be used to solve the same level problem, i.e., they would be allowed to deal with the world with different degrees of knowledge abstraction.

3. Lower Stage Tasks

It is defined as Lower Stage Tasks (LST), the following activities:

a. *Pre-Processing Level: sensing, filtering and distribution subsystems:* -At this level, the activities to be performed are: data acquisition, camera control mechanism and early signal processing. It will extract data from the image and will distribute only the desired information to the appropriate modulus of the second level.

b. *Low Level: early processing; responsible for object extraction:*
-At the level of object extraction, we will have the Target Extraction Module (TEM) that will feature the early image processing system. This subsystem will be characterized by highly parallel and robust algorithms.
This level will provide as output, segmented image in several interesting regions as well as the contour of rigid objects with all surfaces reconstructed.

c. *Mid Level: secondary processing; responsible for object recognition and tracking:*
-This level is composed of two subsystems: Target Recognition Module (TRM and Target Tracking Module (TTM).

The aim of the Target Recognition Module (TRM) is to provide target identification information to the High Level Reasoning Module (HLRM), target orientation information to the Target Tracking Module (TTM) and region inhibition information to the Target Extraction Module (TEM). The Target Tracking Module (TTM) will provide estimate of target acceleration using more refined information from the imaging system and use this to improve the estimate of future target motion.

4. High Level Processing - Reasoning

Responsible for information fusion and coordination, maintenance of world knowledge and user interaction. At this level will be attempted to have "human-like" capabilities added to the system. In order to achieve this will be incorporate extensive data bases and knowledge processing into our system to simulate human expertise as well as will be provided mechanisms for direct human intervention. The action at this upper-level can be subdivided into two main areas: High Level Tracking Module and High Level Reasoning Module.

4.1. High Level Tracking Module (HLTM)

- This subsystem goals will be achieved by the use of "intelligent predictors" designed to incorporate knowledge of the target tactics and objectives, and disposition of friendly forces to help predict likely paths for targets. The use of battlefield knowledge as well as tactical data incorporated into the system, will be utilized to find priority and select the target to be tracked.

4.2. High Level Reasoning Module (HLRM)

- The Reasoning Module (HLRM) is the one responsible for:

i. information fusion from then lower level algorithms and the intelligent predictor;

ii. maintenance of world knowledge, enemy capabilities and tactics;

iii. coordination of the lower level algorithms and judgement of their performances;

iv. coordination of sensors performance as well as their focus of attention in order to extract extra identification information;

v. learning capability of target characteristics and tactics;

vi. evaluation of threat and action sequence based on tactical rules and intelligence.

A detailed description of the above subproblems could be shown as:

i . Information Fusion

The HLRM will be receiving data from lower level algorithms such as: identification, position, predicted trajectory, orientation, velocity, acceleration and confidence values generated by these algorithms. In this process, data coming sensors are used in parallel for feature extraction at the lower levels of the hierarchical structure here proposed. Same or different features are extracted in parallel by these algorithms and fed up to the Reasoning Module. The knowledge based understanding process will then unify these preprocessed data through multilevel representation, combining quantitative and symbolic attributes [Pau-89] as well as a posteriori probabilities.

An extensive bibliography on Sensor Fusion and Reasoning with Uncertainty will be utilized in order to better solve this problem. Among those we can list the following:

[1] -Brooks, R.(1981)- "Symbolic reasoning among 3-D and 2-D images"-Artif. Intellig., 17, pp. 285-349;

[2] -Berenstein, Kamel and Levine (1986)-"Consensus Rules", in *Uncertainty in Artificial Intelligence* - Ed. Kanal & Lemmer - North Holland, pp. 27-32;

[3] -Chen, S.(1987)-"Multisensor fusion and navigation of mobile robots"- Int. J. Intell. Syst., 2, pp. 24-36;

[4] -Chiu, S.L. et all (1986)-"Sensor data fusion on a parallel processor"- Pro.. IEEE Conf. on Robotics and Automation, San Francisco, April, pp. 1629/33;

[5] -Cohen, P.R.(1985)- *"Heuristic Reasoning about Uncertainty: an Artificial Intelligence approach"* -Pitman, Boston, MA;

[6] -Dubois and Prade (1988)-"Representation and Combination of Uncertainty with Belief Functions and Possibility Measures"- Comput. Intell.,4, pp. 244-264;

[7] -Felix and Hoffman (1989) - "A Knowledge-based system using reasoning with uncertainty in the VLSI chip synthesis domain"-Eng. Applic. of AI, 2, June, pp. 144/152;

[8] -Halpern and McAllister(1989)-"Likelihood, Probability and Knowledge"- Comput. Intell., 5, pp. 151-160;

[9] -McArthur, G.-(1988)-"Reasoning about knowledge and belief: a survey"- Comput. Intell., 4, pp. 223/243;

[10] -Pau, L. (1989)-"Knowledge representation approaches in sensor fusion"- Automatica, 25, 2, pp. 207-214.

ii . Maintenance of World Knowledge, Enemy Capabilities and Tactics

As we will be dealing with problems related to real-time operation of algorithms we will be faced with the task of representing and reasoning from real knowledge rather than facts and rules. We will not be dealing just with the problem of understanding knowledge but abstracting, classifying and structuring this knowledge.

Some suggested literature in the area are:

[11] -Broeders, Bruijn and Verbruggen (1989)-"Real-time direct expert control using progressive reasoning"- Eng. Appl. of AI, 2June, pp. 109-119;

[12] -Caudill, M. (1987-1989) -"Neural Network Primer"-AI Expert-Part I: Dec.87, pp. 46-52; Part II: Feb. 88, pp. 55-61; Part III: June 88, pp. 53-59; Part IV: Aug. 88, pp. 61-67; Part V: Nov. 88, pp. 57-65; Part VI: Jan. 89;

[18] -Hood and Mason (1986)-"Knowledge-based systems for real-time applications"- *Applications of Expert Systems* -J.Rao Quinlan-Ed., v.1, pp. 102-119;

[19] -Rao, Jiang and Tsai (1989)-" Combining symbolic and numerical processing for real-time intelligent control" - Eng. Appl. of AI, 2, march, pp. 19-27;

[20] -Simmonds, W.H. (1988)-"Representation of real knowledge for real-time use" - Proc. IFAC on Artif. Intell. in Real-Time Control, Swansea-UK, pp. 07-11;

[21] -Voss, H. (1988)-" Architectural issues for expert systems in real-time control"- Proc. IFAC on Artif. Intell. in Real-Time Control, Swansea-UK, pp. 01-06.

iii. Coordination of the Lower Level Algorithms and Judgement of their Performances

- The algorithms included in this task are those utilized for: Target Extraction Module (TEM), Target Recognition Module (TRM), Target Tracking Module (TTM) and the Intelligent Predictor.

Various algorithms will be associated in the processes of Target Extraction, Recognition and Tracking, some more simple and some others more sophisticate, some utilizing the same data and extracting the same features or complementary features.

In order to achieve "harmony" in the operation of the overall system, knowledge based systems will be utilized. The tunning of the various algorithms that are going to be processed in parallel fashion, with different computing time, different accuracies and different data is a difficult task.

Will be utilized an extensive bibliography already known as well as new development will be required. As a start, the development of the "subsumption architecture" proposed by [Rosemblat and Payton-89] will be taken into consideration.

We can not forget that the operation of these algorithms are in real-time, therefore we ought to decide what AI approach will be more appropriate to deal with this kind of problems. Whether it will be "expert system", or "neural networks", or a "hybrid combination of the two", is in study right now.

Some bibliography in use are:

[22] -Arzén, K-E. (1988)-" An architecture for expert system-based feedback control"- Proc. IFAC on Artificial Intell. in Real-Time Control, Swansea-UK, pp. 21-26;

[23] -Brooks, R (1986)-" A robust layered control System for a mobile robot" - IEEE J. Robotics and Automation, RA-2,1,march, pp. 14-23;

[24] -Brooks and Flynn (1989)-"Rover on a chip"- Aerospace America, October, pp. 22-26;

[25] -Giralt, Chatila and Vaisset (1983)-" An integrated navigation and motion control system for autonomous multisensory mobile robots"- *Robotics Research I* - Brady and Paul, Eds. - Cambridge, MA, MIT, pp. 191-214;

[26] -Handelman and Stengel (1988)-" Perspectives on the use of rule-based control"- Proc. IFAC on Artif. Intell. in Real-Time Control, Swansea-UK, pp. 27-32;

[27] -Kanayama, Y. (1983)-" Concurrent Programming of intelligent robots"- Proc. IJCAI, pp. 834-838;

[28] -Krijgsman, Verbruggen and Bruijn (1988)-"Knowledge-based real-time control"- Proc. IFAC on Artif. Intell. in Real-Time Control, Swansea-UK, pp. 13-19;

[29] -Rosemblat and Payton (1989)- "A fine-grained alternative to the subsumption architecture"- Submitted to: 1989 IEEE/INNS- Int. Joint Conf. on Neural Nets.

iv . Coordination of Sensors performance as well as their Focus-of-Attention

-This subproblem relates to the control signals that are going to be sent to the sensors. These signals will generate focus-of-attention signals to the sensors and trackers to concentrate on smaller areas of the image and the zoom lens camera to track the target for extra identification information.

The sensor performance can be measured by the contribution they can bring into the system through split weighted factors for each sensor individually. Extensive literature on filtering can be used for this evaluation of performance.

v . Learning Capability of Target Characteristics and Tactics

This item will be studied together with the Intelligent Predictor.

vi . Evaluation of Threat and Action Sequence based on Tactical Rules and Intelligence

This item will be studied together with the Intelligent Predictor.

5. Concluding Remarks

Studies on the High Level Reasoning Module have just begun. There is a large amount of additional work to be done, such as a more complete bibliographical search, conceptual design and detailed development. We will continue building our expertise in the field of Artificial Intelligence and to explore ways to use this technology in the Hierarchical Target Recognition and Tracking system.

6. Figure-1: "The Proposed Solution"

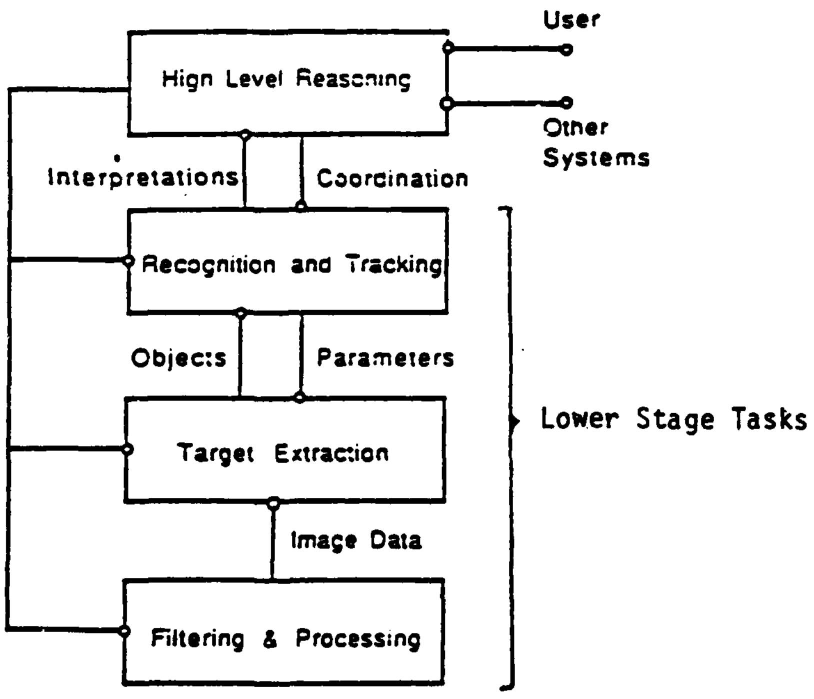


Figure 1: The Hierarchical System Description

Appendix E
The Intelligent Predictor

Author: Jimmy Krozel

1. Introduction

In this appendix, a statement of the intelligent predictor problem, subproblems, and a discussion of how they relate to classical artificial intelligence problems in planning is discussed. First, the intelligent predictor problem is presented with motivation from the ground vehicle tracking scenario. The theory needed for intelligent prediction is dependent on some subproblems from a "dual" domain of problems, namely, from the field of path planning of autonomous vehicles. Thus, a portion of this appendix is devoted to discussing fundamental problems in path planning, and presenting the "dual" problems for prediction. Applications are given to clarify the connection between the planning and prediction problems.

2. Intelligent Prediction

Let us consider the short term objective of a land vehicle in a battlefield environment. The driver has some mission objective that influences the general decisions made in controlling the vehicle to some destination. Physically, the vehicle is constrained to the surface of the earth. However, there are certain features of the terrain that the vehicle cannot or might not traverse. Rocks, ravines, lakes, trees, debris, high slope regions, and other obstacles are features maneuvered around during the course of driving the vehicle to some destination. The influence such environmental conditions have on the decisions made by the driver are a complex combination of human factors, driving experience, mission objectives, tactics, etc., and developing a relationship between the environmental causes and the drivers actions may not be feasible, if even possible. What is desired in intelligent prediction is to develop a model for a decision making program that, given the same environment, produces similar actions compared to the actual vehicles motion. The distinction here is that the decision making program may not necessarily make its decisions based on the same factors as the human driver, however, it is designed to produce decisions that are close to the actual drivers decisions.

The intelligent predictor uses "world data" to predict future motion. As describe above, the battlefield environment is complex; short term motion of the order of seconds may be influenced by visibility conditions, terrain features, mission objectives, tactics, intelligence reports, etc. In order to predict the future motion of the vehicle, one must model these types of data, and reason about how they influence the motion. The intelligent predictor can use a map model of the world (e.g., a DMA map database) to see what terrain the tracked vehicle will encounter in the future. It also has available to it intelligence information about possible strategic locations that the vehicle might be headed for as a destination. It is this type of "world data" that the intelligent predictor is trying to add to the estimation problem in order to intelligently bias the estimation of where the vehicle will be in the future.

The general problem statement for intelligent prediction is made based on the requirements of the tracking problem. The vehicle position and velocity can be estimated at a given instant, and it is desired to predict where it will be k seconds in the future (k from 1 to 10 seconds is practical). Of all the locations that the vehicle can be in k seconds, one desires to order these locations based on the probability (according to some measure) that the location will be the actual location in k seconds. Determining how the vehicle is influenced by environmental conditions can be based on observing successive data of the vehicles position and velocity. The observed vehicle motion can be used to best fit a decision objective function for the intelligent predictor algorithm. For the intelligent predictor algorithm, if the decision objective function were to be applied to the initial estimated position and velocity, then the resultant position and velocity determined by the intelligent predictor algorithm should be very similar to the actual motion. The same objective function is then used with "world data" in front of the vehicle to determine the most probable locations that the vehicle might be at in k seconds. In general, the intelligent predictor reasons about the past actions and the future "world data" to determine the future actions of the vehicle. In doing this, the intelligent predictor uses any "world data" that might influence the future motion of the vehicle terrain features ae an example.

Intelligent prediction may be considered as the "dual" of intelligent planning. Intelligent planning is a well researched topic of autonomous land vehicle control, where the objective of the control system is to guide the vehicle safely from one location in the environment to another in some intelligent manner. For instance, the objective may be to go from one point to another using the shortest path [LP,SS], the fastest route [DX], the least visible route from an observer [M1,M2], or the safest route [OY]. Such routes are considered intelligent because they attempt to produce efficient motion in terms of fuel considerations, safety, time constraints, or any mission objectives modeled by an objective function. An actual driver told to obey the same mission objectives may produce similar actions compared to the results determined by intelligent planners. In planning then, one is given a start location, a finish location, and an objective function to minimize. The path that minimizes the objective function is considered the solution, the intelligent motion. The "dual" problem for intelligent prediction is similar. The start location is known (determined through estimation), but the objective function and the destination location is not known. Thus, in the prediction problem one seeks to establish the objective function and the destination location based on observed data of position and velocity.

In the next sections, several fundamental planning problems are presented. The solution techniques for these problems are briefly discussed, and "dual" problems in prediction are given. Applications are given to make the connection between the planning and prediction problems clear.

3. Road Network Problems

Consider a simple planar graph search problem: find a least cost path in a graph from some start vertex to a finish vertex. This is a standard graph theory problem that is the core search problem to many intelligent planning problems. The graph considered may be a standard graph or a directed graph (digraph). For this problem, there are two standard solution search techniques from graph theory: Dijkstra's Algorithm [Di] and the A^* Algorithm [N,HNR].

The Planar Graph Search Problem is the fundamental problem in graph theory that results when solving road network problems. Graph theory can be directly applied to solve prediction problems for vehicles constrained to roads. Such a prediction problem results when a vehicle is observed to be on a road, and the vehicle is assumed to remain on the road in the future. Being constrained to roads is modeled as being constrained to the graph representing the road network. The intelligent prediction problem analyzes the graph to determine what positions on the road network the vehicle can be in the future. Road traversability can be accounted for, and situations like slowing down for corners or turns can be modeled in the graph.

4. The Obstacle Avoidance Problem

Consider the Obstacle Avoidance Problem [Y,M2,SSH,ScS]: plan a path from a start location to a finish location for a vehicle constrained to motion in a plane, but forbidden to pass through obstacle regions. The obstacles for this problem can be conveniently represented by polygon obstacles, which can be recorded by polygon vertex locations. Alternatively, obstacle regions can be specified by a binary array.

Solution techniques for the Obstacle Avoidance Problem depend on the representation of the terrain and on the optimality criteria. A standard grid [M2] can model possible paths, and a search algorithm can specify a minimum distance path. This technique can produce finely detailed paths; a quadtree [M2] (octree if in 3D) can be used to eliminate unnecessary detail in the solution path, if coarser solutions are sufficient. If the solution path must obey a criterion of maintaining some reasonable amount of clearance from obstacles, then the generalized cone [KZB] or configuration space [L,LW] technique can be used. The Voronoi diagram [Dr,LD] of the obstacles is used in the retraction technique [OY] to solve this problem when maximum obstacle clearance is required. The visibility graph technique [LW] can be used to find the minimum distance path around polygon obstacles.

Obstacle Avoidance methods can be used in several ways to help predict vehicle motion. If one assumes that certain features of the terrain can be considered as obstacles, and that the tracked vehicle driver is influenced by these obstacles, then one can investigate the paths that are most likely to be followed by an obstacle avoiding vehicle. Determining the parameters used for obstacle clearance is then part of the job of the intelligent predictor, as well as determining a probability that a given point is the destination point of the vehicle. The Voronoi diagram technique gives a convenient topological analysis of paths around obstacles, thus allowing a prediction system to generalize about paths in the environment.

5. The Discrete Geodesic Problem

Consider the Discrete Geodesic Problem [M1]: plan the shortest path from a start location to a finish location for a vehicle constrained to motion on a surface. The surface may be represented as a polyhedral surface (facets, edges, vertices), may be represented by a map of contour lines (iso-elevation curves), or may be an elevation array (e.g., DMA land elevation data).

The solution technique for the Discrete Geodesic Problem involves an unfolding of the terrain surface [M1]. The search technique is called the continuous Dijkstra technique, and can generate optimal solutions [M1].

The Discrete Geodesic Problem is relevant to the intelligent prediction problem when the vehicle is in a hilly terrain environment. If the terrain is not hilly, then the Obstacle Avoidance Problem is more appropriate, since it approximates the terrain to be a plane. A similar "dual" prediction problem results compared to the obstacle avoidance prediction problem. The intelligent predictor must determine a probability that a given point is the destination point of the vehicle, assuming that the vehicle follows a discrete geodesic path.

6. Weighted Region Problem

Consider the Weighted Region Problem [M1]: plan the least cost path from a start location to a finish location for a vehicle constrained to motion in a plane of weighted regions; the cost to traverse a region is proportional to the weight. Regions of equal cost can be represented by polygon regions (convex regions are required) or in array form.

Solution techniques for the Weighted Region Problem depend on the degree of approximation used [M1]. The optimal solution is found using a technique that applies Snell's Law from physics and a continuous Dijkstra search technique. This solution may require a complicated search, and alternatively, one can apply an approximate solution technique. The region graph approach gives general results by considering paths that progress from region to region, without analyzing their entry and exit points to regions. Also, a grid search graph technique parallels the same approach, but in finer detail.

The Weighted Region Problem is related to the intelligent predictor problem. If one wants to exploit the feature based representation of terrain data and the effects it has on the vehicle motion, then the weighted region problem can describe paths through the different feature regions. For example, a feature map may be given that specifies the type of land cover for the terrain: low grass, tall grass, sand, gravel, pavement, water, ravines, bushes, trees, etc. One can estimate the traversability of these surfaces, and consequently establish a weight to each region. Then the weighted region prediction problem attempts to analyze the regions that the observed vehicle has passed through in order to predict the regions that the vehicle will pass through in the future.

7. Summary

The intelligent predictor problem was introduced as a problem of predicting the destination of a vehicle in the future by establishing an objective function and probable destination locations based on observed data of position and velocity. The intelligent predictor attempts to add "world data" to the estimation problem, so that knowledge like terrain features indicated by a terrain map and intelligence reports of strategic destination locations can also be be used to determine the most probable future locations for the vehicle. Classical artificial intelligence problems in planning were discussed as the "dual" domain of planning problems; thus, allowing us to consider the results of planning research for the prediction problem.

8. References for Appendix E

[B] Brooks, R.A., "Solving the Find-Path Problem by Good Representation of Free Space," *IEEE Trans. on Systems, Man, and Cybernetics,* Vol. SMC-13, No. 3, pp. 190-197, Mar./April, 1983.

[C] Canny, John, *The Complexity of Robot Motion Planning,* Ph.D. Thesis, MIT, May, 1987.

[Di] Dijkstra, E. W., "A Note on Two Problems in Connection with Graph Theory," *Numerische Mathematik,* Vol. 1, pp. 269-271, 1959.

[Dr] Drysdale, R.L., III, *Generalized Voronoi Diagrams and Geometric Searching,* Ph.D. Thesis, Stanford University, Stanford, CA, Jan., 1979.

[DX] Donald, B. and Xavier, P., *A Provably Good Approximation Algorithm for Optimal-Time Trajectory Planning,* Tech. Report TR88-947, Dept. of Computer Science, Cornell Univ., Ithaca, NY, Nov., 1988.

[HNR] Hart, P.E., Nilsson, N.J., and Raphael, B., "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Trans. of Systems, Science, and Cybernetics*, Vol. SCC-4, No. 2, pp. 100-107, July, 1968.

[KZB] Kuan, D., Zamiska, J., and Brooks, R.A., "Natural Decomposition of Free Space for Path Planning," *IEEE International Conference on Robotics and Automation*, IEEE Computer Society, St. Louis, MO, pp. 168-173, March, 1985.

[L] Lozano-Perez, T., "Automatic Planning of Manipulator Transfer Movements," *IEEE Transactions of Systems, Man, and Cybernetics*, Vol. SMC-11, pp. 681-89, Oct., 1981.

[LD] Lee, D.T. and Drysdale, R.L. III, "Generalization of Voronoi Diagrams in the Plane," *SIAM Journal of Computing*, Vol. 10, No. 1, pp. 73-87, Feb., 1981.

[LP] Lee, D.T. and Preparata, F.P., "Euclidean Shortest Paths in the Presence of Rectilinear Boundaries," *Networks*, Vol. 14, pp. 393-410, 1984.

[LW] Lozano-Perez, T. and Wesley, M.A., "An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles," *Communications of the ACM*, Vol. 22, No. 10, pp. 560-570, 1979.

[M1] Mitchell, J.S.B., *Planning Shortest Paths*, Ph.D. Thesis, Stanford University, Stanford, CA, Aug., 1986.

[M2] Mitchell, J.S.B., "An Algorithmic Approach to Some Problems in Terrain Navigation," *Artificial Intelligence*, Vol. 37, Nos. 1-3, pp. 171-201, Dec., 1988.

[N] Nilsson, N.J., *Principles of Artificial Intelligence*, Tioga Pub. Co., Palo Alto, CA, 1980.

[OY] O'Dunlaing, C. and Yap, C.K., "A 'Retraction' Method for Planning the Motion of a Disc," *Journal of Algorithms*, Vol. 6, pp. 104-111, 1985.

[ScS] Schwartz, J.T. and Sharir, M., "A Survey of Motion Planning and Related Geometric Algorithms," *Artificial Intelligence*, Vol. 37, Nos. 1-3, pp. 157-169, Dec., 1988.

[SSH] Schwartz, J.T., Sharir, M., and Hopcroft, J., (Eds.), *Planning, Geometry, and Complexity of Robot Motion*, Ablex Pub. Corp., Norwood, NJ, 1987.

[SS] Sharir, M. and Schorr, A., "On Shortest Paths in Polyhedral Spaces," *SIAM Journal of Computing*, Vol. 15, No. 1, pp. 193-215, Feb., 1986.

[Y] Yap, C.K., "Algorithmic Motion Planning," *Advances in Robotics*, Schwartz, J.T. and Yap, C.K. (Eds.), Lawrence Erlbaum Associates, Hillsdale, NJ, pp. 95-143, 1987.

Appendix F
Navigation and Path Planning for Autonomous Aircraft: Voronoi Diagram Approach

By: Krozel, J., Andrisani, D.

(This paper has been accepted for publication in the *AIAA Journal of Guidance, Control, and Dynamics*)

Navigation Path Planning for Autonomous Aircraft: Voronoi Diagram Approach

Jimmy Krozel* and Dominick Andrisani II†
Purdue University, West Lafayette, Indiana

Introduction

FOR autonomous aircraft and intelligent pilot aids, navigation path planning may be performed by computer rather than by humans. In this Note, we consider the task of planning a path from some start location to some finish location in mountainous terrain. The technique investigated employs searching for the best paths among topologically unique paths. Candidate paths are depicted by a search graph generated using a computerized geometric construct.

In reviewing approaches to navigation path planning, one must consider path-planning research involving robot manipulator arms, mobile robot and autonomous land vehicle path planning, and existing planning algorithms for pilot aid and autonomous aircraft.[1-6] For instance, the Dynapath algorithm[3] and Beaton et al.[4] generate path plans in fine detail using tree branching and grid-based approaches, respectively. We do not use a grid approach. However, a search graph is constructed using a Voronoi diagram so that only *one* path between each neighboring mountain is represented. This represents a higher level of abstraction for the path-planning problem in our work compared to Refs. 2 and 3. A second-stage path refinement is suggested for the paths generated with our technique.

Voronoi diagrams have also been used in path-planning problems in robotics.[5] In Voronoi-based methods, the Voronoi diagram of obstacle polygon line segments is used to create a diagram of straight and parabolic arcs connected to form a search graph. Meng[6] uses such a technique for autonomous aircraft path planning. We use the Voronoi diagram of a set of points, rather than a set of line segments, to generate a search graph for path planning.

Problem Statement

The simplified navigation path-planning problem addressed here is to find a path from a start location S to a finish location F on a mountainous terrain map while minimizing the path length and the exposure to threats. The problem considered here is constrained to a constant altitude. Mountains are described by a terrain contour map and threats are described by danger regions on the contour map. For any segment of a trajectory that crosses a threat area, it is assumed that a cost associated with danger can be assigned to that segment.

Constructing a Voronoi Diagram Search Graph

A path-planning search graph depicts possible paths through the terrain/threat search space. A good search graph includes at least one path between all neighboring mountain obstacles. Paths are represented within a graph of nodes connected by arcs, where the arcs represent possible paths to be flown. The search environment can be separated into two spaces: free space and obstacle space. Free space, through which the vehicle is free to move, is unoccupied by obstacles, and obstacle space is occupied by obstacles. We will proceed to model free space with a feasible search graph, a graph including only nodes and arcs that are in free space.

Frequently, the planning objective is to minimize the path length and threat exposure. For this purpose, path length and threat cost are weighted and summed according to their relative importance; a final cost is assigned to each graph arc. This requires that the threat cost can be expressed in terms of the cost of path length.

The Voronoi diagram of a set of points is used for the purpose of creating a search graph for path planning. First, the definition of a Voronoi diagram is given to introduce this geometric construct. Then, with a simple example, the use of Voronoi diagrams for constructing terrain search graphs is presented.

A Voronoi diagram for a set of N points p_i, $1 \leq i \leq N$, in the Euclidean plane is a partitioning of the plane into N polygonal regions, one region associated with each point p_i. Figure 1 shows the Voronoi diagram for a set of points. A point p_i is referred to as a Delaunay point. The Voronoi region $V(p_i)$ associated with point p_i is the locus of points closer to p_i than to any of the other $N-1$ points. The Voronoi edge separating $V(p_i)$ from $V(p_j)$ is composed of the points equidistant from p_i and p_j. Not all Voronoi edges are bounded; some extend to infinity. The intersection of Voronoi edges occurs at vertices called Voronoi points. The construction of Voronoi diagrams is reviewed in Refs. 1 and 7.

A procedure is now presented for creating a search from a Voronoi diagram. Given a terrain contour map, we first polygonize the contour data, as shown in bold lines in Fig. 2. Vertices from mountain polygon obstacles are used as Delaunay point locations to model obstacles. The vertices of obstacles are points that should be avoided when traversing around polygon obstacles. Voronoi edges, by definition, maximally avoid these points. A Voronoi diagram is constructed for the set of Delaunay points describing obstacle vertices. The Voronoi edges that extend infinitely are bounded by a region that encloses the entire terrain map, creating path search arcs around the boundry of the terrain map. Figure 2 illustrates such a Voronoi diagram. The complete Voronoi diagram is shown in dashed and solid edges. Next, Voronoi edges defined by two neighboring Delaunay points of the same obstacle are removed. These edges are shown in dashed lines. Start and finish search nodes can be added to the search graph by connecting these point locations to the closest Voronoi edge. The resulting search graph is described by the remaining solid edges.

This method gives a good representation for free space using a fairly small amount of nodes. Note that only one path results between neighboring obstacles. Thus, all paths depicted by the search graph are topologically different. Another salient feature of this method is that it guarantees that the resultant search graph will depict only feasible paths, which are completely in the free space. Feasible paths are guaranteed based on parameter of the polygonization: Let the closest distance from any vertex of an obstacle to a neighboring obstacle be defined as δ. If obstacle polygonization is performed using polygon sides no greater in length than δ, then the resultant Voronoi search graph will depict only feasible paths. A proof of the feasibility of search graphs generated with this method is given in Ref. 1.

Navigation Path-Planning Example

Consider a set of polygons that represent mountains at a constant altitude. Figure 3 shows a set of mountain polygon obstacles as solid black regions. The vertices of polygon mountain boundaries are used to generate the search graph illustrated. Graph arcs from map boundaries eliminate any arcs from infinite Voronoi edges, and start and finish nodes are added to the graph by connecting them to the closest Voronoi point nodes. The threat environment consists of a barrier of threats so that no path free of threat exists between the start and finish nodes. Dashed circular regions indicate threat regions.

A solution path should have minimal exposure to threats. To account for this, arcs of the search graph are assigned a threat cost in addition to a length cost. For this example, the cost assigned to each arc in the search graph is the Euclidean distance between the nodes forming the arc. Within threat regions arcs are assigned a threat cost in addition to the length cost. The threat cost is either 3 or 10 times the cost for traversing an arc of the same length in a nonthreat region. These levels of danger are indicated in parentheses as $(3\times)$ and $(10\times)$ in Fig. 3. In addition to these threat regions, a constraint is added so that the solution path does not use the boundary arcs to pass around the barrier of threats. This is established by assigning a cost of 100 times the length cost for using an arc on the boundary. This is labeled as $(100\times)$ in Fig. 3.

The Voronoi diagram search graph of Fig. 3 is searched to find the minimum cost path from the start node S to finish node F. The A^* algorithm[8] is used for this purpose. The A^* search technique uses problem-dependent information to reduce the number of nodes investigated. The heuristic function[8] is the Euclidean distance to the finish node F. This is an admissible heuristic,[8] and thus, the results of the A^* algorithm are optimal. The optimal path plan is shown with bold edges in Fig. 3. Since no path free of threats exists, the solution path penetrates the barrier of threats in the least dangerous region. Notice that the optimal path penetrates two $(3\times)$ threat regions. The first region is the least costly location to penetrate the initial barrier of threats. The second region is penetrated because the alternative path around this region contains a boundary arc in the search graph. Boundary arcs are heavily weighted, and thus the path through the $(3\times)$ region results.

Path Refinement

The method described provides a solution path from the start point to the finish point. While this path is guaranteed to avoid mountains, it may not represent the best path when compared to other topologically similar paths. For example, the resultant path may not penetrate individual threats at their weakest points. The next step in path planning may be to consider solutions near the Voronoi diagram path. One possible procedure for searching for path plans near the Voronoi diagram path is to search a fine grid in the region around this path. Another possible procedure is to use an iterative improvement procedure such as the path relaxation method of Thorpe.[9] Finally, when refinement is performed further information may be included in the cost function. For example, the cost function may account for the aircraft's performance capabilities or limitations. Large changes in heading angle may be punished, as well as paths that come too close to mountain boundaries.

Conclusions

A technique for generating a search graph depicting topologically unique paths around mountain boundaries at a constant altitude is presented. Our technique requires a description of mountain boundaries as polygons and generates the search graph using a geometric construct. All nodes and arcs of the search graph are guaranteed to lie in free space, thus avoiding mountain obstacles. A solution path plan is generated by searching the graph for the optimal path from a start location to a finish location.

Acknowledgments

This research was conducted at the NASA Ames Research Center, Moffett Field, California, under NASA Grant NCC 2-367. The first author is currently supported by the Artificial Intelligence Center, Hughes Research Laboratories, Malibu, California, as a Hughes Doctoral Fellow.

References

[1]Krozel, J. A., "Search Problems in Mission Planning and Navigation of Autonomous Aircraft," M. S. Thesis, Purdue Uiversity, West Lafayette, IN, May 1988.

[2]Mitchell, J. S. B., "Planning Shortest Paths," Ph.D. Thesis, Stanford University, Stanford, CA, Aug. 1986.

[3]Denton R., Jones, J. E., and Froeberg, P. L., "Demonstration of an Innovative Technique for Terrain Following/Terrain Avoidance, The Dynapath Algorithm," *IEEE National Aerospace and Electronics Conference*, Dayton, OH, May 1985, pp. 522–529.

[4]Beaton, R. M., Adams, M. B., and Harrison, J. V. A., "Real-Time Mission and Trajectory Planning," *IEEE Conference on Decision and Control*, Los Angeles, CA, Dec. 1987 pp. 1954–1959.

[5]Schwartz, J. and Yap, C. K., *Advances in Robotics*, Lawrence Erlbaum Associates, Hillside, NJ, 1986, pp. 187–228.

[6]Meng, A., "Flight Path Planning Under Uncertainty for Robotic Air Vehicles," *IEEE National Aerospace and Electronics Conference*, Dayton, OH, May 1987, pp. 359–366.

[7]Lee, D. T. and Schachter, B. J., "Two Algorithms for Constructing a Delaunay Triangulation," *International Journal of Computer and Information Sciences*, Vol. 9, No. 3, June 1980, pp. 219–242.

[8]Nilsson, N. J., *Principles of Artificial Intelligence*, Tioga Publishing, Palo Alto, CA, 1980, pp. 53–97.

[9]Thorpe, C. F., "Path Relaxation: Path Planning for a Mobile Robot," The Robotics Institute, Carnegie-Mellon University, Pittsburgh, PA, CMU-RI-TR-84-5, April 1984.

Fig. 1 Voronoi diagram for a set of 20 points.

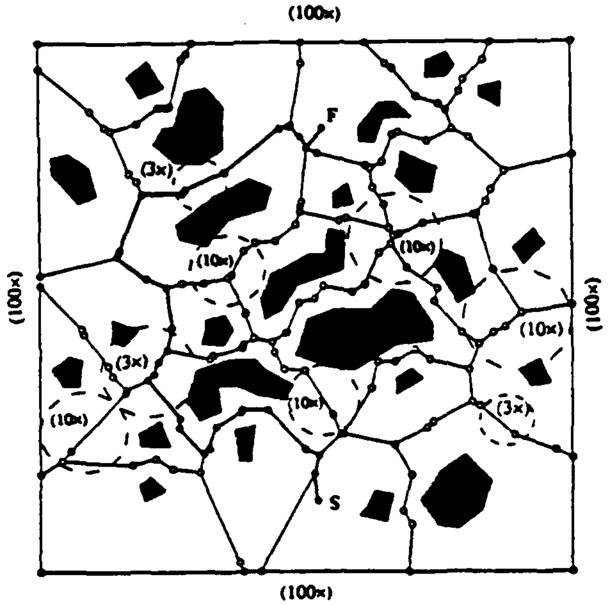
Fig. 2 Voronoi diagram graph for polygon obstacles modeled with Delaunay points at the vertices of the obstacles.

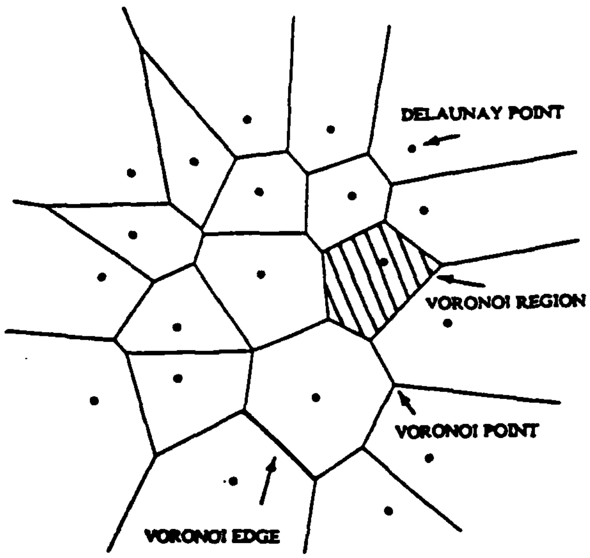
Fig. 3 Voronoi search graph for a mountainous terrain environment with threats.

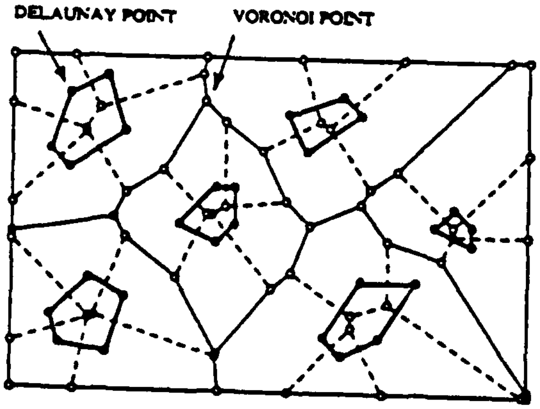
Received May 26, 1988; revision received Feb. 5, 1989.

*Hughes Doctoral Fellow, School of Aeronautics and Astronautics. Student Member AIAA.

†Associate Professor, School of Aeronautics and Astronautics. Member AIAA.


(100×)
F
(3×)
(10×)
(10×)
(10×)
(3×)
(10×)
(10×)
(3×)
(10×)
S
(100×)


DELAUNAY POINT
VORONOI REGION
VORONOI POINT
VORONOI EDGE


DELAUNAY POINT
VORONOI POINT